DESIGN AND IMPLEMENTATION OF A BIOLOGICALLY REALISTIC OLFACTORY CORTEX MODEL

By

VITOR MANUEL GRADE TAVARES

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2001

To My Wife Cristina and Sons, Samuel and Pedro.

ACKNOWLEDGMENTS

First I would like to thank my advisor Dr. Jose Principe for the quality guidance and friendship he has demonstrated throughout this study. He allowed me the freedom to explore by myself, but was always attentive and pertinent in critical moments. It is really a privilege to work with someone with the background and knowledge he has.

I would also like to thank Dr. John Harris for the help with the VLSI analog implementations. His invaluable comments were an asset to this research. I would like to thank my committee members Dr. Haniph Latchman, Dr. Purvis Bedenbaugh, and Dr. Robert Fox for the interest and precious time they spared participating on my supervisory committee.

I thank Dr. Pedro H. H. Guedes de Oliveira for the friendship demonstrated at the end of this study. He was of great help giving insightful ideas whenever I needed. I thank INESC-Porto for letting me use the facilities and services to test some of my integrated circuits. I also thank to Lucilia Fernandes for the secretarial work that freed me from extra tasks.

I also thank FCT (Fundacao para a Ciencia e Tecnologia) in Portugal for financial support. Without it this Dissertation would not be possible. I want to extend my gratitude to FLAD (Fundacao Luso-Americana para o Desenvolvimento) for financing part of the final phase of this research. Special thanks go the CNEL for their friendship and for creating a good working environment. I especially remember my roommates Raul Pinheiro and Helder Cochofel but also thank Marc Boillot and Jason Nadrowski for the friendship and

help they offered me. I would like to thank Natercia Principe and family for always being helpful whenever I needed. I also thank Miguel Santos for his help with the University paper work when I was away.

My wife, Cristina, has my deepest appreciation for all the suffering she had to endure during this research period. Only a great love would be able to stand a long and involuntary separation, giving the support I needed at the same time. Very special thanks go to my sons, Samuel and Pedro, for the light they switched on in my life. Since they were born a rainy day becomes sunshine whenever I look at them. I also thank the rest of my family for all the support, encouragement and love.

TABLE OF CONTENTS

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

DESIGN AND IMPLEMENTATION OF A BIOLOGICALLY REALISTIC OLFACTORY CORTEX MODEL

By

Vitor Manuel Grade Tavares

December 2001

Chairman: Dr. Jose C. Principe
Major Department: Electrical and Computer Engineering

The major goal of the research presented in this dissertation is to develop an efficient framework composed of a digital signal processing (DSP) and analog CMOS-VLSI implementation platforms to design, simulate and study Freeman's model of the olfactory cortex. Conceptually, this biologically plausible model is a coupled lattice with local stable dynamics but with global nonconvergent behavior. The methodology followed in this research shows that Freeman's model can be efficiently discretized using DSP techniques, providing an alternative to more traditional Runge-Kutta integration methods. An obvious outcome of the system's discretization is that a network of dynamic components can implement the resulting discrete time model, which enhances analysis and optimization of Freeman's model for information processing.

For optimization purposes, the digital computer is more flexible, and allows for immediate parameter setting and topology manipulation. However it is not the most efficient in computational terms. Analog implementations are preferred in terms of round-off noise, power and speed, but bring the problem of determining appropriate parameters. This work proposes a mixed signal (analog amplitude discrete-time) CMOS-VLSI implementation of Freeman's model that is unified functionally with the DSP simulation. Hence, the DSP machine can be efficiently used to find parameter sets that give equivalent responses to the analog system. The fine tuning should be operated on the DSP model and computed at the analog implementation level.

In this dissertation, we propose, fabricate, and test the required building blocks of Freeman's model. The dynamical part of Freeman's model is the hardest to implement. Due to the slow dynamics, it is really a challenge to implement such a system in a small area and be power efficient. A local offset cancellation method is proposed for the gamma kernel, with low sensitivity to the feedback parameter and implemented with switch-current (SI) techniques. A novel class of discrete state filters is developed, called filter and hold (F&H patent pending). The F&H techniques are step invariant and allow the integration of long time constants with reasonably small area. Finally, a multiplexing method is proposed and tested that take full advantage of the F&H innovation, efficiently allocates fewer resources among all channels, and is transparent to the computation being implemented. Several chips were fabricated and measured to prove the operationality of each proposed block, and to demonstrate that both the DSP and analog chips behave as predicted.

CHAPTER 1
INTRODUCTION

1.1 Introduction

This dissertation develops an efficient framework, as well as analog and digital implementation platforms, to facilitate the study and design of Freeman's model of the olfactory cortex. Conceptually, this biologically plausible model is a coupled lattice with local stable dynamics but with global nonconvergent behavior. Instead of modeling the olfactory cortex at the neuron level, Freeman [1–3] has proposed an hierarchical dynamical model that bases its formal representation on the response of thousands of cells called the neural cell assembly. Each cell assembly is described by a second order dynamical equation. This level of description has been called mesoscopic to mean an intermediate level between global states that lead to behavior and single neurons. The functional and anatomical connections between cell assemblies are also modeled, and the result is a complex dynamical system built from basic second order nonlinear elements in a hierarchy of levels called Katchalsky [1] and designated by K0, KI, KII and KIII. At a first glance, Freeman's model of the olfactory system may look similar to artificial neural networks (ANN) found in engineering, which are also arrangements of static or dynamic PEs (processing elements) [4–7]. However, unlike ANN topologies, in the olfactory cortex there is a very tight relation between topological arrangement of neurons and function, that is preserved in the Katchalsky set. Furthermore, Freeman reports that the computational paradigm is based on nonconvergent dynamics [3] (i.e., dynamical systems that have singularities higher than second order—chaotic attractors [8–11]) . This marginally stable system may be able to respond rapidly with high sensitivity and specificity to minute changes in the input, and recover effortlessly while waiting for the next input. Conversely, a system exhibiting point attractor dynamics (such as the Hopfield nets [12,13]), needs large input energy to begin a new search. Thus, it seems that dynamical systems with higher order singularities, namely the ones that exhibit nonconvergent dynamics, comply with rapid response to input changes due to the sensitivity to initial conditions, allowing quick trajectory divergence from small input energy differences. This is very different from what has been done in neural networks and machine learning. Freeman's model breaks with the logic of fixed-point terminal dynamics, advancing a new computational paradigm for information processing. But it also raises many difficult questions such as, how is information represented in these nonconvergent systems, what is the importance of nonconvergent dynamics in information processing, and how can we retrieve information from the resulting complex time series? The answer to these questions will bring an all new perspective on information and signal processing. However, to answer these questions, more flexible and efficient computational platforms are needed, and this is the motivation behind this dissertation.

1.2 Olfactory Cortex Model

1.2.1 Population Models

Neurons are the discrete components that form a neural network [14]. They are interconnected in a structure to perform some higher level operation. A neuron can accept inputs from thousands of other neurons and conversely drive as many others. However, Freeman has stated that information processing transcends the neuron level. Information processing instead is a group property because, in sensory cortices neurons are for the most seen as having random firing patterns [15,16]. Thus, the appropriate scale to model information processing in the brain, is the mesoscopic level. Instead of single neurons, Freeman's model classifies populations of neurons as continuous functions representing the main connections in the population called a neural set. Furthermore, it is also possible to classify the principal connections between the sets. The neural network then takes the form of a network of sets.

Neural and network sets are functional structural level representations of physical networks. The descriptive approach of neurons and networks as functional structures imply the introduction of functional connections. These are functions that describe relations between neural sets. In this perspective we have to interpret them as mesoscopic models when compared to the microscopic individual neurons. The same applies to the states referred to as mesoscopic-states representing the state of a neuron population. These functions represent the relations between the set states (mesoscopic-states).

Although individual neurons have outputs represented by discrete spikes, the density of spikes measured in a population of neighborhood neurons can be modeled as a continuous variable [1,2]. A mesoscopic state emerges in neural sets as continuous activity density functions, varying both in space (depends on the neuron's relative position on a neural mass), and in time (integration performed at the neuron's membranes). This is true under the assumption that for vanishingly small regions, the neighborhood neurons in that region share similar inputs and so have similar states. The activity density functions are derived from neuron models and measurements made in living, but anesthetized animal brains [1]. These models are better described in terms of dynamics and are expressed in two modes: a wave mode, representing the pulse to wave transformation in the set; and a pulse mode, representing the output activity of the set.

Depending on the main type of functional interconnections between neurons in a set, a topological hierarchy of sets can be built. Because sets are represented by dynamical models, the network of sets can become extremely complex and intractable when described as a whole. To make the model more amenable, the complexity is wrapped in a hierarchy embedding simpler structures that are repeated as the network grows into more complex topologies. The hierarchy levels in increasing order of complexity are designated by K0, KI, KII, KIII (K stands for Katchalsky due to his pioneer work [1]) and will be described below. From an engineering perspective the hierarchical model is also very appealing because it allows the same basic building blocks to be used and reused.

1.2.2 The K0 model

The simpler structure in the hierarchy of sets is shown in Figure 1-1 and is called the K0 set. The neural population in a K0 neural set, shares common inputs and a common sign for the output: either positive for excitatory neurons, or negative for inhibitory neurons, and is the basic building block of all networks. The model can accept several spatial inputs that are weighted and summed. This input is then processed by a 2nd order linear time invariant system, defined by Eq. (1-1).

$$\frac{1}{a \cdot b}\left(\frac{\partial^2}{\partial t^2}x(t) + (a+b)\frac{\partial}{\partial t}x(t) + a \cdot bx(t)\right) = f(t) \qquad (1\text{-}1)$$

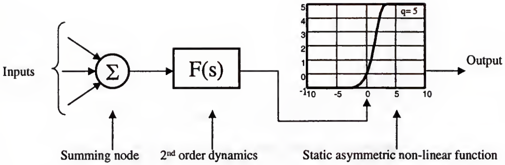


Figure 1-1. The K0 model

Equation (1-1) together with the weighted sum of the input, forms the pulse-to-wave density activity function where the integration is performed. This function is empirically obtained from averaged measurements done in a specific brain region of an anesthetized animal [17]. Measurements are taken from multiple trials as an average of the extracellular neural potential. It reflects the many dendritic currents taken as an average from many neurons in the region [2]. The filtered wave signal is then reconverted into a pulse-wave signal. The conversion is nonlinear and is defined in the olfactory model by Eq. (1-2). The K0 symbol is shown in Figure 1-2 and Figure 1-3 shows an example of a K0 network set.

$$Q(x(t), q) = \left.\begin{cases} q(1-e^{-(e^{x(t)}-1)/q}) & \text{, If } x(t) > x_0 \\ -1 & \text{, If } x(t) < x_0 \end{cases}\right|_{x_0 = \ln(1-q\ln(1+1/q))} \qquad (1\text{-}2)$$

Figure 1-2. The K0 symbol

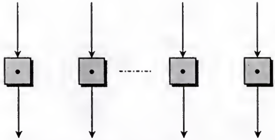


Figure 1-3. K0 network set

1.2.3 The KI Model

The KI set is the second model in the hierarchy. It can be built from K0 cells that interact through lateral feedback (auto-feedback is not allowed) of common sign: positive for excitatory KI sets or negative for inhibitory KI set (Figure 1-4).

A KI network set is made of fully interconnected KI cells shown in Figure 1-5. The interconnecting weights vary both in space and time. Nevertheless, it is sufficient to define the lumped circuit model where the feedback weights are fixed (i.e., it means that the subject is in a given physiological stage, e.g., awake or asleep). The interconnectivity is defined by Eq. (1-3) where $f_i(t)$ represent the forcing input in Eq. (1-1) (I is an input vector and the indexes stand for cell position in the network).

$$\begin{bmatrix} f_0(t) \\ f_1(t) \\ \dots \\ f_i(t) \\ \dots \\ f_{N-2}(t) \\ f_{N-1}(t) \end{bmatrix} = \frac{1}{N} \begin{bmatrix} 0 & K_{pp} & \dots & K_{pp} & \dots & K_{pp} & K_{pp} \\ K_{pp} & 0 & \dots & K_{pp} & \dots & K_{pp} & K_{pp} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ K_{pp} & K_{pp} & \dots & 0 & \dots & K_{pp} & K_{pp} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ K_{pp} & K_{pp} & \dots & K_{pp} & \dots & 0 & K_{pp} \\ K_{pp} & K_{pp} & \dots & K_{pp} & \dots & K_{pp} & 0 \end{bmatrix} \cdot \begin{bmatrix} Q(x_0, q) \\ Q(x_1, q) \\ \dots \\ Q(x_i, q) \\ \dots \\ Q(x_{N-2}, q) \\ Q(x_{N-1}, q) \end{bmatrix} + I \qquad (1\text{-}3)$$

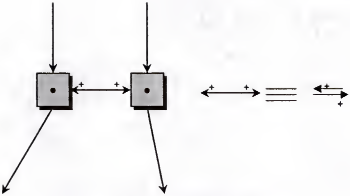


Figure 1-4. Excitatory KI set model

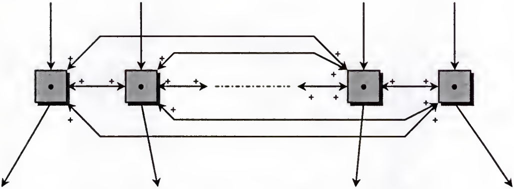


Figure 1-5. Excitatory KI network set

1.2.4 The KII Model

The third level of the Katchalsky hierarchy is the KII model. This is the most important building block of the overall olfactory system. The KII set can be built from two KI sets, interacting through both excitatory and inhibitory connections. Figure 1-6 is a topological representation of a KII set that is used to model the olfactory bulb (Section 1.2.5). The KII set is defined by Eq. (1-4) (Figure 1-6 for indexes).

$$A = \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ f_4(t) \end{bmatrix} = \begin{bmatrix} 0 & K_{MM} & -K_{MG} & -K_{MG} \\ K_{MM} & 0 & -K_{MG} & 0 \\ K_{GM} & K_{GM} & 0 & -K_{GG} \\ K_{GM} & 0 & -K_{GG} & 0 \end{bmatrix} \cdot \begin{bmatrix} Q(x_1, q)) \\ Q(x_2, q)) \\ Q(x_3, q)) \\ Q(x_4, q)) \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} I \qquad (1\text{-}4)$$

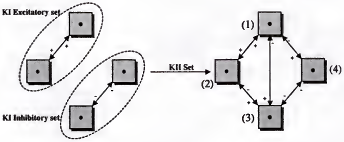


Figure 1-6. The KII set model

The KII set has a peculiar behavior. The response to an impulse is a damped oscillation. However, with a sustained input it evolves into an undamped oscillation which vanishes only after the input returns to zero. Therefore, the KII set is an oscillator controlled by the input.

Figure 1-7 shows a representation of a KII network set. Considering functionality, the KII network is an arrangement of mutually coupled oscillators. With a more careful look into Figure 1-7, two main types of interconnections can be discerned: excitatory through the top K0 cells and inhibitory through the bottom K0 cells. The inhibitory connections are fixed while the excitatory ones are trainable. By strengthening excitatory connections the coupling increases. With the input presence, stronger coupled sets will build more energy oscillation than the others. At the same time, they will inhibit even more the other sets from building energy through the inhibitory connections. Thus, the excitatory connections are responsible for cooperative behavior while inhibitory connections introthe channels [17] (a channel is an input or output taken from a single set). In this perspective the KII network is an associative memory that associates an input pattern with the spatial distributed coupling that is stored in the interconnections [17,18].

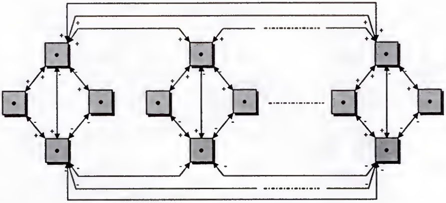


Figure 1-7. The KII network set

For "on-off" type of input patterns, the learning can be accomplished with a modified Hebbian rule as follow [17]. If between any two input channels the product $I_j \cdot I_i$ (i, j are channel indexes) is equal to "on" (both "on"), then the connections between the corresponding channels are increased to a high value, otherwise the value is kept unchanged. At the beginning with no learned patterns, all these connections are set at a low value. This method effectively unbalances the in-between channels coupling to accomplish the pattern representation.

1.2.5 The KIII Model—The Olfactory System

The final Katchalsky level is the KIII network and represents the olfactory model. The KIII model is a variant of the KII network with several layers of these elements connected through linear delay functions.

The full model is depicted in Figure 1-8 and the names for the different layers and KO cells, represent real cortical regions and cells, of a mammalian brain. The inputs are collected at the receptors in the nose and are represented in the model by a set of inputs. These inputs are then fed into the periglomerular layer (PG) made up of periglomerular (P) cells and into the mitral cells (M) at the olfactory bulb (OB). Mitral cells are the origin of all the signals for the rest of the olfactory system and they interact with granular cells (G) to form the KII oscillators.

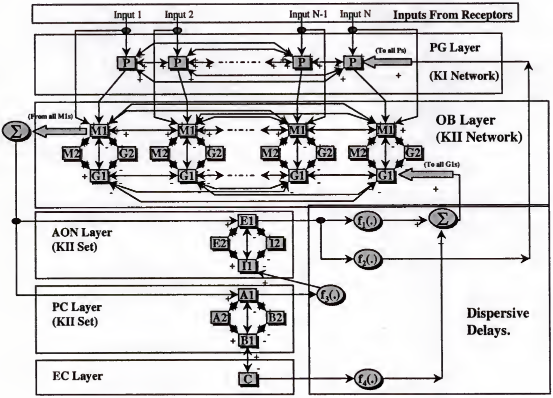


Figure 1-8. The KIII network set—the olfactory system

The output collected from mitral cells excite the anterior olfactory nucleus (AON) and the prepyriform cortex (PC). The PC output goes into the external capsule (EC) by deep pyramidal cells (C) and also disperses into the AON and OB. AON by its turn does the same into the granule cells in the OB and into the PG layer.

The axons that interconnect different regions of the brain, have different thicknesses and lengths, differentiating their electric properties. Consequently, the action potentials arrive at the target with different delays. The effect is a low-pass filtering operation of the axon density pulses [3] and are modeled by the dispersive delay functions $f_j(.)$ shown in Figure 1-8.

The KIII network is functionally similar to the KII network. However, the natural frequencies of the KII cells in the OB layer and the other layers, are incommensurable. This together with the different interlayer dispersive delays interaction, the natural tendency for the oscillators to lock to a single frequency is prevented and nonconvergent, chaotic behavior arises [16]. From the point of view of information processing, the KIII is a highly unstable associative memory. When there is no input excitation, the system state exists in a high dimensional attractor (the basal attractor). This high dynamical dimensionality means that the system can quickly reach any point in its state space, and quickly react to any input. Once the input is applied the system dimensionality is reduced to a lower chaotic attractor where recognition takes place. Once the input disappears, the KIII goes back to the basal state.

1.3 Motivation for the Dissertation

Freeman's model discussed above is biologically realistic, but extremely difficult to understand from the point of view of information processing. Engineering systems are designed to be stable, and therefore avoid nonconvergent regimes. Understanding Freeman's model is therefore crucial to further our knowledge about brain dynamics, and devise man-made systems with similar properties. Freeman's model is normally simulated using high-level numerical integration methods, that are time consuming and not amenable to simple modifications. In this dissertation we propose an equivalent digital model using DSP (digital signal processing) techniques. The result is a system made of components that can easily be subject to alterations. It also brings easier probing, it is computationally more efficient and it is better fitted for real-time simulations because the system is updated on a sample-by-sample basis.

The flexibility, immunity to noise and modularity of the DSP implementation is unsurpassed when compared to analog VLSI implementations. However, there are advantages in analog VLSI that cannot be overlooked such as: speed (only limited by the delay on each component), intrinsic parallel computation, effective infinite magnitude resolution (free of round-off problems), and power efficiency. These benefits motivate the development of an analog VLSI implementation preserving as much as possible the functional correspondence to the digital design. The DSP simulator may then be used to find the right analog VLSI circuit parameters avoiding the burdensome task of finding the appropriate parameters with analog circuit simulators. Our analog design approach is centered on continuous magnitude and discrete-time implementation for the following reasons:

- Continuous magnitude response avoids round-off errors that has been reported to have a negative effect on the simulation of chaotic dynamical systems [19–22].
- The full olfactory cortex model is made of many repeated cells fully interconnected. If a full analog (continuous-time) implementation is followed, the number of resources and area required for the interconnect would be prohibitive. With a discrete-time implementation control over the timing of events is gained. The syn-

chronous update at discrete times provide free time slots for transparent multiplexing with consequent optimal resource management.

- It allows for DSP formal correspondence which is important for parameter search.
- Unexpectedly, time discretization also lead to a very efficient implementation of the slow dynamics called F&H (filter and hold) that allows the integration of long time constants in a reasonably small area.

1.4 Dissertation Outline

Chapter 2 presents the discrete olfactory model system. The results are confronted with reported results available in the literature, and with the normal numerical integration methods. In chapter 3 a local offset cancellation method for gamma kernels is proposed. Gamma kernels implemented with switch-current (SI) techniques, exhibit output offsets that are strongly dependent on the time constant. The technique presented in this chapter minimizes this effect for single ended input and single ended output signals.

The dynamical part of the Freeman model is the hardest to comply with. Implementing the slow dynamics in analog VLSI with small area and power efficiency is a challenge. Chapter 4 proposes a novel class of filters, called filter and hold (F&H), that are step invariant and allow the integration of long time constants with reasonably small area.

Chapter 5 presents all other components such as the nonlinearity and summing nodes, needed to build a K0 cell. Chip measurements for KII cells are presented, and compared with the DSP simulator solutions.

A multiplexing method that efficiently allocates few resources among all channels is proposed. Chapter 6 provides the concluding remarks and discusses future work directions.

CHAPTER 2
DISCRETE NETWORK IMPLEMENTATION OF THE OLFACTORY MODEL

2.1 Introduction

Freeman's model is normally simulated using Runge-Kutta integration [23]. This integration procedure is time consuming and not amenable to simple modifications in the topology. This chapter develops an equivalent digital model using DSP techniques at the system level (most results found in this chapter were published by Principe, Tavares, Harris and Freeman [24] and by Tavares, Principe and Lynn [25]).

Discrete-time system modeling has been successfully applied to physical phenomena which accept a linear model, such as the vocal tract model in speech synthesis [26]. Instead of sampling the outputs of the continuous-time model, this modeling approach discretizes the system equations using the theory of digital signal processing. However, for nonlinear systems there is no known transformation that can map phase space to a sampled domain, like the conformal mapping does from the S to the Z domains [27]. Nonetheless, here we are interested in a very special nonlinear system built from an interconnection of K0 models that possesses linear dynamics followed by a smooth static nonlinearity. It was experimentally verified that the methodology explained in this chapter to design equivalent discrete-time linear systems is successful in preserving the general dynamical behavior of the continuous-time model.

Equation (2-1) shows the general formal structure of the KIII network. The forcing input in the right hand side of Eq. (2-1) represents the connections from other similar K0 sets in the network, which themselves might also receive excitation from this given ith K0 set. The left hand side of Eq. (2-1) represents a LTI (Linear Time Invariant) system.

$$\frac{1}{a \cdot b}\left(\frac{\partial^2}{\partial t^2}x_i(t) + (a+b)\frac{\partial}{\partial t}x_i(t) + a \cdot bx_i(t)\right) = \sum_{j \neq i}^{N} [K_{ij}Q(x_j(t), q) + K_{ij}'f_j(Q(x_j(t), q), t)] + I_i \qquad (2\text{-}1)$$

The proposed procedure to discretize Eq. (2-1) decomposes this equation as topology and dynamics. The dynamics can be mapped with linear DSP techniques to the discrete-time domain. The connecting structure between individual cells being static is preserved, except that delays are incorporated in the feedback pathways to avoid instantaneous time calculations (Figure 2-1). The feedfoward connections do not need to be delayed. There are two basic potential problems with this methodology: first, it assumes that the continuous dynamical system is synchronized, which is unrealistic (although not covered in this work, there are ways to mitigate this shortcoming [28]). Second, the dynamics between the analog and discretized models may differ. Similar methods have been applied to discretize the Hopfield model (with smooth nonlinearities) [12,13,28] and the terminal dynamics of both (analog and discrete) systems have been shown equivalent [29]. Yet, the terminal dynamics for the KII set is a limit-cycle, instead of a fixed point and the inclusion of a delay in the feedback path could change the frequency of oscillation between the analog and the digital counterpart. The implementation discussed here avoids this problem, and experiments have shown that the method is a excellent approximation to Runge-Kutta as long as the nonlinear block is static and that the final dynamics is *1/f* type.

The proposed discretization approach results in a discrete-time model that can be implemented with a *network of dynamical components* found in standard commercial neural network packages or DSP simulators. This means that adding or deleting features or new components to the system is simple and does not imply rewriting the global state equations. The resulting digital system also computes the output solutions in a sample-by-sample basis which is better fitted for real-time simulations. These are invaluable characteristics for testing of the analog simulator presented in the following chapters.

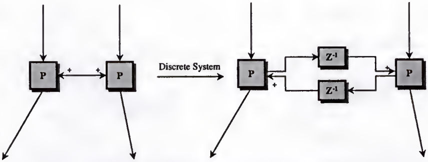


Figure 2-1. Discrete representation of a KI set feedback paths

2.2 Impulse Invariant Transformation

The simplest method to discretize the local linear dynamics of the K0 set of Eq. (2-1) is to make an impulse invariant transformation [30] that preserves the system impulse response shape. This characteristic should, if well controlled, also preserve the global dynamics because the static part (topology) is smooth and is kept unchanged through the mapping. The discrete-time system impulse response, $\hat{h}(nT)$, is found by sampling the continuous-time impulse response, *h*(*t*), with a suitable sampling frequency f_s=1/*T* (Eq. (2-2)). The spectrum of the sampled impulse response is constituted by replicas of the base-band frequency spectrum, centered at multiples of the sampling frequency f_s. Therefore, the method is subject to aliasing, but it can be controlled because both K0 and KIII network dynamics are of 1/*f* type. The transfer function that results from the invariant transformation applied to the left hand side of Eq. 2-1 is defined by Eq. (2-3) where *F*(*z*) is the forcing input..

$$\hat{h}(nT) = T \cdot h(nT) \tag{2-2}$$

$$\hat{H}(z) = \frac{X(z)}{F(z)} = a \cdot b \cdot T \cdot \frac{1}{a-b} \cdot \frac{(e^{-aT} - e^{-bT})}{1 - (e^{-aT} + e^{-bT}) \cdot z^{-1} + e^{-(a+b)T} \cdot z^{-2}} \cdot z^{-1} \tag{2-3}$$

From observation of Eq. (2-3), the impulse invariant transformation already incorporates the delay required to avoid the instantaneous computations caused by the interconnection feedback of the KIII. In this way no extra delay is actually needed because the delay operator found in Eq. (2-3) can be moved to the feedback connections.

2.3 Gamma Basis Decomposition

Alternatively, one can decompose the impulse response of Eq. (2-3) over a set of real exponentials which has been called the gamma bases (also known as the gamma kernels) given by Eq. (2-4) [31,32]

$$g_k(n) = \binom{n-1}{k-1} \mu^k (1-\mu)^{n-k} \tag{2-4}$$

The main objective of such decomposition is to allow easy global optimization of the KII and KIII networks using methods such as backpropagation through time (BPTT) [33], once the input and output signal examples are made available in later stages of the research.

The impulse response of the gamma kernels (a cascade of low-pass filters of equal time constant) resembles many of the signals collected or modeled in neurophysiology (Figure 2-2) [34]. The gamma kernels form a set of complete real basis functions, meaning that any signal in L_2 (finite power) can be decomposed over gamma kernels with an arbitrarily small error [32]. The gamma kernels can be used as delay operators in substitution to the ideal delay operator, bridging very well continuous and discrete-time domains.

Weighting the values of each gamma delay operator creates a generalized feedforward filter called the gamma filter [31], which is written as Eq. (2-5). The vector W=[w_0 w_1... w_N]^T is the projection vector. The gamma filter block diagram is shown in Figure 2-3.

$$x_k(n) = (1-\mu)\cdot x_k(n-1) + \mu\cdot x_{k-1}(n-1)$$
$$y(n) = \begin{bmatrix} w_0\ w_1\ \dots\ w_N \end{bmatrix}\begin{bmatrix} x_0(n) \\ x_1(n) \\ \dots \\ x_N(n) \end{bmatrix} \qquad (2\text{-}5)$$

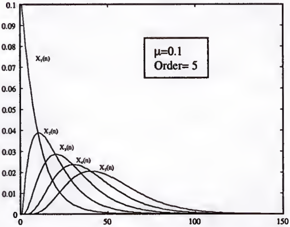


Figure 2-2. Impulse response at different taps of a 5th order gamma filter

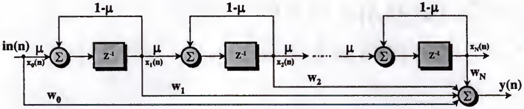


Figure 2-3. Gamma filter structure

Decomposing Eq. (2-2) over the gamma basis follows the well-established system identification framework, as represented in Figure 2-4 [35]. A white noise source is fed to the sampled differential equation (2-3) (noted by $H(z)$). The same signal is applied to the gamma filter $G(z)$. The output of these two systems are subtracted and the resulting value is used as an error for adaptation of the gamma parameters. Adaptation algorithms have been developed previously to search μ and the projection vector W [32].

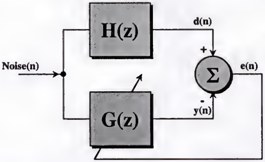


Figure 2-4. System identification (ID) procedure

The filter order determination in the case of Eq. (2-3) is relatively easy because the unknown system is decomposed in the set of real exponentials. To gain a better understanding, the characteristics of the gamma kernel will be addressed next.

2.3.1 Gamma Filter Stability, Resolution and Depth of Memory

Consider the Z transform of Eq. (2-4) for M gamma kernels. The transfer function between the input and the last gamma tap can then be written as:

$$G_T(z) = \left(\frac{\mu}{z-(1-\mu)}\right)^M \qquad (2\text{-}6)$$

Therefore, the delay mechanism for the gamma filter is a cascade of identical first order IIR sections (Figure 2-3). For stability the gamma filter feedback parameter has to be restricted to $0 < \mu < 2$. Yet the condition is very easy to test since μ is common to all kernels in the filter.

The gamma filter has an infinite region of support, which means that a low order filter can model long time dependencies. The region of support of the impulse response is very important for system identification and for processing of time information, it implements a short-term memory mechanism. Memory in the present context means how far in the past the dependencies are to compute the present output. The memory depth for the gamma filter with M taps is [32]:

$$D = \frac{M}{\mu} \tag{2-7}$$

Gamma filters are able to trade memory depth with resolution. In fact, defining M as the order of the filter, D the memory depth given by Eq. (2-7), and R the time resolution, we can write for the gamma filter $M = D \cdot R$. In an FIR filter, the memory depth is the filter order M. From Eq. (2-7) we have the extra feedback parameter μ to change the memory depth of the gamma filter. However, there is a trade-off between time resolution and memory depth, which is controlled by the order of the system. The resolution is problem dependent and most of the time the trade-off is resolved heuristically. In the present dynamic system modeling, the lowest order was selected due to analog VLSI implementation constraints.

2.4 Differential Equation Discretization Results

From biological evidence the parameters *a* and *b* assume the values 220/s and 720/s [1], respectively. Since the poles are real, a good fit can be obtained with a second order gamma filter. The sampling frequency 1/T was chosen as 20 times larger than the maximum frequency pole. Figure 2-5 represents the impulse invariant system response to an impulse. The gamma filter approximation of the sampled differential equation is also plotted. The resulting match is very good, although the approximation filter has only a double pole at (1-μ). Table 2-1 summarizes the gamma filter parameters values found using the discussed system ID approach.

Table 2-1. Gamma parameters

w0	-0.0250
w1	1.0673
w2	7.7234
μ	0.0237

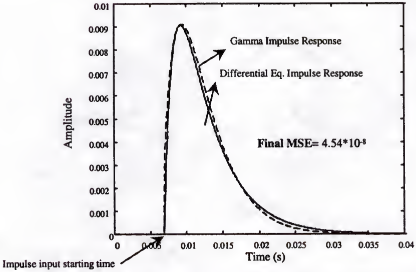


Figure 2-5. Gamma basis approximation of the impulse response of Eq. (2-3)

2.4.1 Dispersive Delays Discretization

As discussed earlier, the time dispersion was introduced in the KIII network to model the different axonal interconnection lengths and thicknesses. These functions can be modeled as in Eq. (2-8) [3]

$$f(Q(x,q), nT) = \frac{1}{T_s - T_e} \sum_{nT-T_s}^{nT-T_e} Q(x,q) \tag{2-8}$$

which can be rewritten in a convolution form as:

$$\begin{gathered} f(Q(x,q), nT) = Q(x,q) \bullet \frac{1}{T_s - T_e}(u(nT-T_s) - u(nT-T_e)) = \\ Q(x,q) \bullet D_d(nT) \end{gathered} \tag{2-9}$$

where $\bullet$ denotes the convolution and $u(nT)$ is the step function. Equation (2-8) represents a low-pass FIR filter in discrete-time and $D_d(nT)$ is recognized as a linear phase filter of type I or II depending if the number of $D_d(nT)$ samples is even or odd, respectively. To be more biologically realistic the filter should have nonlinear phase. A simple low-pass IIR system, such as a first order gamma kernel, has nonlinear phase and so it implements a more realistic dispersive delay.

Time dispersion can be analytically calculated using the memory depth defined in Eq. (2-7). In Eq. (2-9), $D_d(nT)$ can be interpreted as a memory structure with a time depth T_s-T_e+1. Then, considering a gamma filter of some order, the parameter μ can be calculated making $D=(T_s-T_e+1)/T$ in Eq. (2-7). The order of the gamma dispersive function will impact on resolution. From experimental results, a second order gamma filter (W=[0 0 1]^T) is shown to be sufficient. The resulting dispersive gamma functions impulse responses are plotted in Figure 2-6.

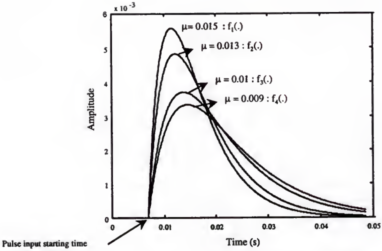


Figure 2-6. Delay response for different dispersive delays of the KIII model

2.4.2 NeuroSolutions Example

After discretization, any software package (DSP or neural networks) or even hardware simulators can be utilized in the simulation. The environment used for this task is a commercial neural network package called "NeuroSolutions" [36]. The software has all the basic building blocks that are needed to construct the Katchalsky sets and the dispersive delays. The user interface enables the creation of the topologies for each set using an icon-based interface. Figure 2-7 shows the KII setup breadboard in NeuroSolutions where the different graphical icons can be identified by the labels.

NeuroSolutions is an object-oriented programming environment for neural networks that is based on a data flow model of computations. There are two basic families in Neurosolutions: the axons (bubble icons in Figure 2-7) where the nonlinearities and short-term memories are implemented, and the synapses where the sum of products are calculated (interconnecting elements). The sum of products can be extended over time giving rise to adaptive filters. By interconnecting different axons and synapses, the user is actually giving indications to NeuroSolutions how to build the code and run it. In each icon is defined the number of elements that it contains (e.g., for the nonlinearity we would define 4 elements for a single KII channel or 16 for four channel KII). Interconnecting different axons through synapses one is able to build a network. Because our discrete-time system is made of components those components can be represented by an icon (axon or synapse) in NeuroSolutions. For instance, to implement the KII set, equations (1-2), (2-5) and the structure shown in Figure 1-6 are needed. In these equations we have a second order recursive filer that can be implemented in NeuroSolutions as the gamma axon with two taps (the gamma delay) followed by a synapse (the weights), and feedback (implemented by another synapse linking the output of the filter to its input). The output of the dynamic part is passed through a nonlinearity that can be implemented by another axon. The program provides the most common nonlinearities found in neural networks, but does not have a nonsymmetric nonlinearity required by Freeman's model. However, the nonlinear axon can be configured to any nonlinearity by writing a very simple DLL (dynamic Link Library). So we programmed the nonlinearity of Eq. (1-2). Once the KII set is created, we need a KII network, which is obtained by fully interconnecting KII sets. NeuroSolutions enables us to create vectors of processing elements, so we have just to define how many KII sets are needed. The iconic representation does not change, but the feedback connectivity is sparse (hence the arbitrary synapse must be used). In order to implement the fully interconnectivity, we have to use another synapse connecting the output of each KII to the input of all the others. The input to the network is implemented by an axon with a wave generator, to the left of the gamma delay line. Figure 2-7 shows how these blocks are laid on the breadboard. NeuroSolutions allows for probing of the activities anywhere in the network. In this case we place a data buffer and a scope at the output of the network to observe the output of the KII network (the output of the filter), once an input is applied.

Now adding or removing KII cells to this system is straight forward and is only a question of network and program parameter setting (e.g., number of repeated cells). To simply loading the parameters of complex breadboards we utilized another feature of NeuroSolutions that allows the loading of weights from an external file.

It should be noted that almost no programming was required to set up the breadboard and once again, this is possible because of our system level discretization where the final model can be dealt as a standard digital artificial neuron network (ANN) model. This Neurosolutions example shows clearly the flexibility gained with a discrete implementation. A 20 channel or a 200 channel network uses the same breadboard because only the indication of how many elements and the corresponding parameter values are needed to actually change the network.

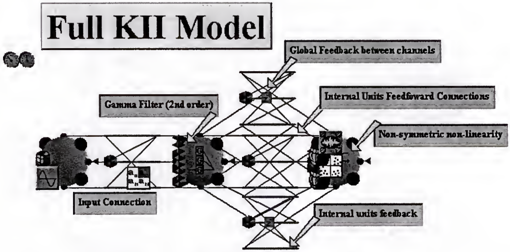

Full KII Model
Global Feedback between channels
Internal Units Feedfoward Connections
Gamma Filter (2nd order)
Non-symmetric non-linearity
Input Connection
Internal units feedback

Figure 2-7. The KII NeuroSolutions breadboard

2.4.3 Digital Simulation Results

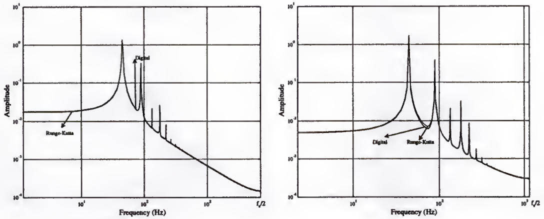
Figure 2-8 shows a comparison of the KII output spectrum to a "high" input obtained using Runge-Kutta (ODE45 in Matlab [37]) and our modeling method. The KII system response was simulated with an input square wave with an amplitude of 6 (arbitrary units) when excitation is present, and zero otherwise. The parameter set is summarized in Table 2-2 where K_{MG} are the weights between the mitral (M) and granular cell (G) denoted by M and G in Figure 1-8.

Table 2-2. KII Parameter set

K_{MM}	0.3
K_{MG}	5
K_{GM}	0.2
K_{GG}	0.25

The magnitude spectrum was computed using the window method. For Figure 2-8a, we took 6 seconds simulation time with a window of 12,000 points, 20% overlap and 11 averages. For Figure 2-8b we performed a 19 seconds simulation time with 2,048 points window, 20% overlap and 40 averages. As can be observed, the match between the Runge-Kutta spectrum and our methodology is very good.

Table 2-3 summarizes the number of flops taken by both methods (DSP and Runge-Kutta) to compare the computational efficiency. It is clear that the DSP technique outperforms the higher level differential equation solver. It should be emphasized that these results are for a free running Runge-Kutta method (i.e., the program is allowed to automatically adapt the best time step). If the same time resolution is forced on both methods then the differences become considerably more noticeable.



a) b)

Figure 2-8. FFT comparison of KII output spectrum using Runge-Kutta and our discretization method. a) $T = 1/(720 \times 20)$; b) $T = 2 \cdot \pi/(720 \times 20)$

Adding extra delays on the feedback paths alters the signal's frequency content. In dynamical terms, an extra delay (or more) changes the order of the system. This change has obvious repercussions on the frequency spectrum since the system poles will move. A simulation example is shown in Figure 2-9. The consequence of a single extra delay is a frequency shift.

Table 2-3. Number of flops comparison

DSP Sampling time	$T = 1/(20 \cdot 720))$	$T = 2\pi/(20 \cdot 720))$
Simulation Time	6 seconds of data	6 seconds of data
Runge-Kutta	15 Million flops	15 Million flops
DSP	12 Million flops	1 Million flops

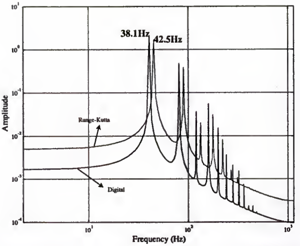


Figure 2-9. DSP result with two delays in the feedback paths

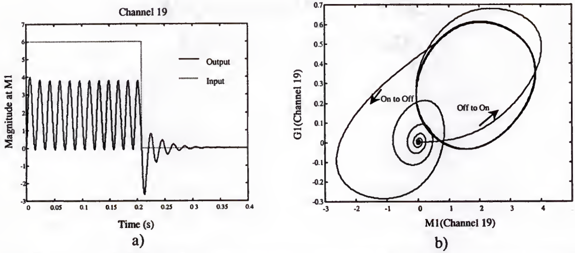
2.4.4 The KII and KIII Network Results

A simulation is performed for a N=20 channels KII network with 5 patterns stored (Table 2-4), set according to the modified Hebbian rule discussed earlier. The results shown are for the recall of pattern 5.

Table 2-4. Patterns Stored

Pattern 1	1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0
Pattern 2	0 0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0
Pattern 3	0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
Pattern 4	0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0
Pattern 5	0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 1 0 1

Figure 2-10a shows that an input excitation drives the system into a limit-cycle that lasts while the input is applied. Figure 2-10b shows the system trajectory with the changing input. As the input rises, the system's trajectory moves away from the origin to a large limit-cycle centered afar from the origin. It stays with "circular" shape for the duration of the input, and returns to zero when the input vanishes. The phase plot resembles a Shilnikov singularity, but the local characteristics of the unstable point need to be investigated further [10].



a)

b)

Figure 2-10. Time series of KII with a square input and phase plot. a) Response to a pulse; b) Phase plot

Figure 2-11 shows the 20 channel KII network outputs after a noisy representation of pattern 5 (14th and 15th bits flipped) is applied (now the input has a maximum magnitude of 2). As expected, channels that have a "1" at the input oscillate with larger magnitude. However, channel 14 can be still recognized as a "1" in spite of the fact that at its input a "0" was applied. If only the steady-state oscillation energy is considered, and a correct energy threshold is used, channel 15 can be set as a "0". Hence, the right pattern 5 association is retrieved even with the noisy input. The information output is then coded in the AM oscillation that Freeman describes in the OB of the rabbit following a sniff.

The interesting property of this output is that it is transient in time, disappearing when the input vanishes. This behavior is very different from Hopfield networks, where the system dynamics stay in the fixed point after the input is removed. Extra energy would have to be supplied to the system to get away from the fixed point, while here the KII network is always ready for real time processing of temporal patterns.

The KIII network was constructed in NeuroSolutions with only N=8 channels in the OB layer because now the simulation takes much longer to find the solutions. Table 2-5 summarizes the parameter set used for the simulation. All the results were generated for the recall of pattern 2. The input is a smooth rising pulse within the values of 2 and 8 as specified in [3].

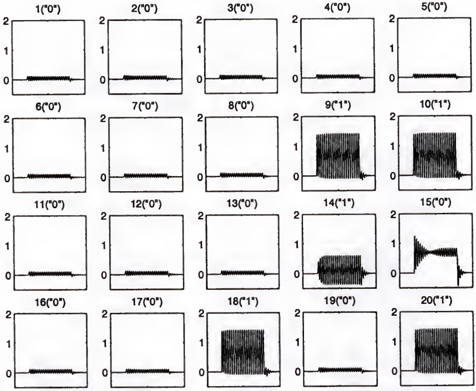


Figure 2-11. KII 20 channel network response to input pattern 5 (Channel response("Input"))

Table 2-5. Parameters set for the KIII network simulation (Figure 1-8)

External Interconnections	$[K_{MP}; K_{EM}; K_{AM}; K_{CB}; K_{BC}]=$ $[0.5; 1.0; 1.0; 1.5; 1.0]$ $[K_{PP}; K_{MM-H}; K_{MM\_L}; K_{GG}]=$ $[0.2; 4; 0.5; 1.5]$
OB Layer KII	$[K_{MM}; K_{MG}; K_{GM}; K_{GG}]=$ $[0.25; 1.5; 1.5; 1.8]$
AON Layer KII	$[K_{EE}; K_{EI}; K_{IE}; K_{II}]=$ $[1.5; 1.5; 1.5; 1.8]$
PC Layer KII	$[K_{AA}; K_{AB}; K_{BA}; K_{BB}]=$ $[0.25; 1.4; 1.4; 1.8]$
Gains Associated with dispersive delays (K_l Associated with $f_l(.)$)	$[K_1; K_2; K_3; K_4]=$ $[1.6; 0.5; 2; 1.5]$
T_e of equation 13	$[T_{e1}; T_{e2}; T_{e3}; T_{e4}]=$ $[11; 15; 12; 24]$
T_s of equation 13	$[T_{s1}; T_{s2}; T_{s3}; T_{s4}]=$ $[20; 26; 25; 39]$
Patterns Stored	$\begin{bmatrix} Pattern1 \\ Pattern2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$

In the full KIII system we have many possible probe points. Let us start by analyzing the resting state at the PG, OB, AON and PC layers. If the time signals are observed at any of these locations, in the absence of input, random fluctuations of a quasi-periodic activity will be noticed resembling the electroencephalogram's background activity. When the activity is plotted as phase plots between the inhibitory and excitatory states (Figure 2-12), we see phase plots that resemble strange attractors of different shapes across the four layers. Therefore the KIII seems to have a high-order singularity producing nonconvergent dynamics compatible with a chaotic attractor. Simple power spectrum estimation of the time series generated at the AON (E1 state) shows clearly a *1/f* type of spectrum that is once again compatible with a self similar attractor (Figure 2-13).

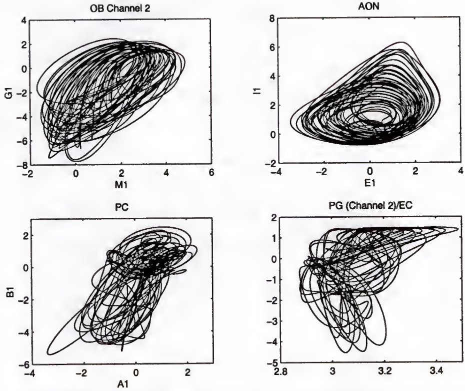


Figure 2-12. Phase plots at different layers of the KIII network (no excitation)

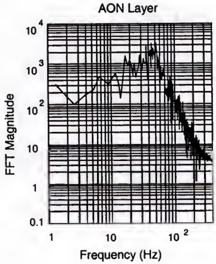


Figure 2-13. AON fast fourier transform (FFT) with no excitation

Let us now turn to the OB layer and analyze the response to a pulse input. Figure 2-14 shows the time series for the recall of pattern two. Notice that the response of the channels where the "high" level of the input was stored do indeed show a DC offset during the time the pulse is applied. An increase in oscillatory amplitude during the pulse was also expected. A system optimization using real data should find better parameters solving this problem. When a phase plot is created using the excitatory M1 and inhibitory G1 states of channel 2, it is clear (Figure 2-15a) that the basal attractor (for the "off" input) changes to a "wing" when the input switches to a high amplitude ("on" input). Channels that correspond a stored pattern, oscillate in perfect synchronization (Figure 2-15b and c) and share the same basal state evolution. This is also reported by Freeman [3]. However, channels belonging to different templates oscillate with different incommensurable phases (Figure 2-15d). It should be referenced however, that in the beginning of the simulation they start in phase as shown in Figure 2-16, but lose synchrony very quickly thereafter. Freeman states that this lack of coherence is not apparent in the signals collected from the rabbit bulb nor in analog implementations of the model. He states that it corresponds to the breakdown of the chaotic attractor possibly due to quantization errors that create limit-cycles in phase space [38]. This behavior indicates, that when excited, the KIII system undergoes a transition from the basal to a different nonconvergent behavior characteristic of the pattern being recalled. When the excitation vanishes the system always returns to the initial attractor (Figure 2-15). The "on" channels are distinguishable by the higher offset and signal dynamics (Figure 2-15). All these results are consistent with Freeman's reports using Runge-Kutta numerical methods [3], while here we use our discretization of the linear part of the model dynamics.

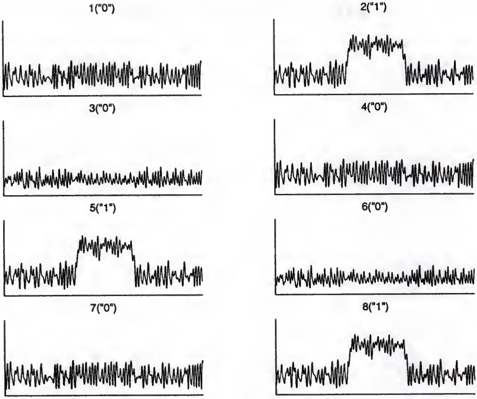


Figure 2-14. Time series for the recall of pattern 2 [channel ("input")]

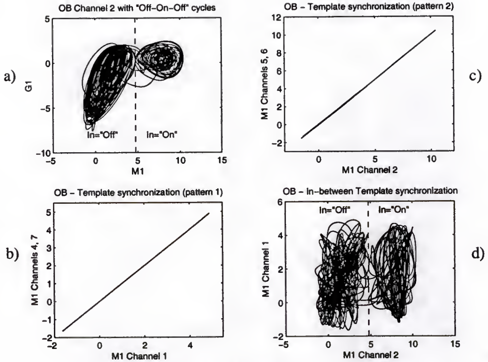


Figure 2-15. Behavior with input "Off-On-Off" cycles

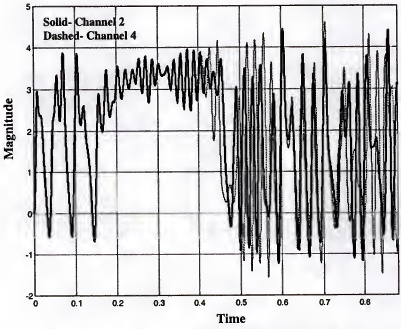


Figure 2-16. OB channel 2 and 4 time series

2.5 Quantization Effects: Some Insights

The previous sections discussed a discrete-time mapping method for the olfactory model using DSP techniques. The resulting digital version utilized the same parameters as the continuous-time model and provided identical results as the ones reported earlier by Freeman using the Runge-Kutta methods. Nevertheless, Freeman has expressed concern relative to the stability of chaotic attractors caused by quantization intrinsic to digital implementations [19,20]. Round-off noise (amplitude quantization) is reported to be the biggest source of instability. The phenomenon is related with attractor crowding [21,22] which occurs when the attractor "size" is in the order of the least significant bits of a computer word length. To express better what is involved, let us take an example and consider a widely used chaotic system formulated in (2-10) and called the Lorenz system [8].

$$\begin{cases} \frac{\partial}{\partial t}x(t) = \sigma(y(t)-x(t)) \\ \frac{\partial}{\partial t}y(t) = r\cdot x(t)-y(t)-x(t)\cdot z(t) \\ \frac{\partial}{\partial t}z(t) = x(t)\cdot y(t)-b\cdot z(t) \end{cases} \qquad (2\text{-}10)$$

The σ, b, r >0 and are parameters of the system. Let us use σ=10, b=8/3, r=28 [11] which produce chaotic dynamics. Euler integration is used to find the solutions. Although Euler numerical integration is a poorer approximation than Runge-Kutta, it is simpler to implement and control. Equation (2-11) is the numerical state space representation of (2-10) using the transformation $\frac{\partial}{\partial t}x(t) = \frac{x(n+1)-x(n)}{T}$.

$$\begin{cases} x(n+1) = x(n)+T\cdot\sigma\cdot[y(n)-x(t)] \\ y(n+1) = y(n)+T\cdot[r\cdot x(n)-y(n)-x(n)\cdot z(n)] \\ z(n+1) = z(n)+T\cdot[x(n)\cdot y(n)-b\cdot z(n)] \end{cases} \qquad (2\text{-}11)$$

The value T was chosen to be 0.01. From the simulation of Eq. (2-11) and measuring the states maximum and minimum values, a certain number of quantizing levels were created. This is a way to bring the numerical precision up to the size of the attractor. The quantizing levels will define the different possible values that Eq. (2-11) mathematical operations may assume. Taking now into account the quantization function *quant*(.) equation (2-11) takes the form of Eq. (2-12).

$$\begin{cases} x(n+1) = x(n) + T \cdot quant[\sigma \cdot quant[y(n) - x(t)]] \\ y(n+1) = y(n) + T \cdot quant[quant[r \cdot x(n)] + quant[-y(n) - quant[x(n) \cdot z(n)]]] \\ z(n+1) = z(n) + T \cdot quant[quant[x(n) \cdot y(n)] - quant[b \cdot z(n)]] \end{cases} \tag{2-12}$$

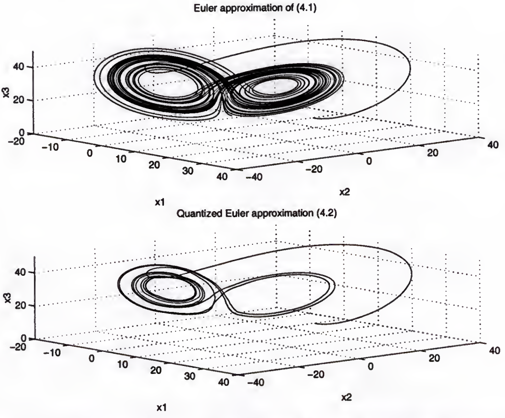
Factors directly related with time dynamics are not quantized. This prevents the system to be to strongly coupled to the quantization. All simulations have an initial condition defined as [*x*,*y*,*z*]= [0 30 0].

Figure 2-17 represents two phase plots. One for a solution to Eq. (2-11) and the other for Eq. (2-12) quantized with 8 levels. Figure 2-18 shows *x*(*t*) time series. The simulation results show clearly that under quantization, a long term non-predictive system like (2-12) can become of low order, namely limit-cycle.

We further tried 32 quantization levels. The results are shown in Figure 2-19 and Figure 2-20. It shows now that a nonconvergent system can become asymptotically stable. In this case the quantization has created a strong basin around the zero fixed point.

Quantization changes dynamics by introducing extra nonlinear effects. So the more "analog" the mathematical operations resemble the more the system "disorganizes" and then more chaotic is its behavior. Quantization seems to be a way to control chaos by limiting the number of possible trajectories in state space. The experiment shows that in a nonlinear system the coarse quantization should not be considered as an independent noise source [30], but instead needs to be modeled as an extra nonlinear function embedded in the original system function.

Figures 2-17 to 2-20 demonstrate that the trajectory gets trapped in some region during its history. This quantization phenomenon has been also reported with the KIII model [20]. One solution that seems to stabilize the chaotic attractor is to add white noise to specific points of the network [19].


Euler approximation of (4.1)

Quantized Euler approximation (4.2)

Figure 2-17. Comparison of a "non" quantized with a 8 level quantization Lorenz system phase plots

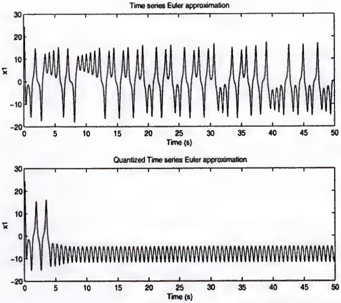


Figure 2-18. Comparison of a "non" quantized with a 8 level quantization Lorenz system time series

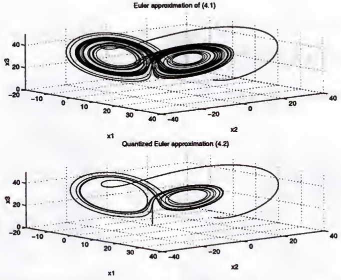


Figure 2-19. Comparison of a "non" quantized with a 32 level quantization Lorenz system phase plots

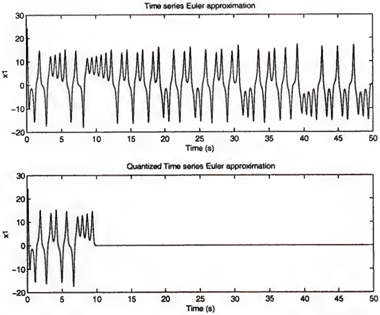


Figure 2-20. Comparison of a "non" quantized with a 32 level quantization Lorenz system phase plots

Taking into consideration the above results we conclude that a discrete-time system that uses full analog calculations will be realistic and exempt of attractor crowding problems. Furthermore, according to previous KIII reported results, the natural noise inherent to any real system will actually work as a chaotic "stabilizer" [19].

2.6 Analog VLSI Implementation—What to Gain?

Round-off noise inherent to digital computers is, in principle, a source of problems for complex systems composed of many coupled oscillators. By itself the round-off noise can be reduced as much as required using more and more bits, but eventually such procedure becomes impractical. An obvious solution is analog computing. Due to the natural effective infinite amplitude resolution the round-off problem is immediately solved. An electronic analog implementation has further intrinsic noise that can help stabilize the chaotic attarctors. Furthermore the computation is parallel which consequently will be more efficient. Nevertheless, in terms of VLSI implementation, the KIII network represents a real challenge, mainly because the number of connections increases proportional to N^2 (N-number of channels). They can actually grow so big to turn impractical the design of a chip. To overcome the massive interconnecting network, multiplexing can be employed to scale down the number of interconnections proportional to N.

In a continuous-time system the use of multiplexing implies time discretization of signals. In order to keep a continuous-time resolution, the output of the multiplexers must be filtered, which represents an extra VLSI implementation difficulty. Our approach is to depart from continuous-time system implementation into a mixed discrete/continuous-time processing framework. In this environment the magnitudes are fully analog, the time resolution can be controlled, and multiplexing comes very naturally.

2.7 Summary

This chapter presented a DSP mapping procedure to model the olfactory system. The impulse-invariant transformation, although a very simple method, is subject to aliasing. However, since the local dynamics are low-pass, a suitable sampling rate minimizes the aliasing effect. Furthermore, impulse-invariant preserves the impulse response of the original filter which is important if the signal's shapes are to be preserved. Another beneficial feature of the transformation was the input delay element. This delay was placed at the feedback connectivity level sparing the use of extra delays with the consequent frequency shift. Results have shown that the DSP method is more computationally efficient than Runge-Kutta with similar outcomes. It also opens new directions for analog hardware implementations.

CHAPTER 3
FEEDBACK INDEPENDENT OFFSET CANCELLATION SCHEME FOR SI GAMMA KERNELS

3.1 Introduction

Switch-current circuits [42,43] are a possible method to avoid low voltage problems through operation in the current domain. The low voltage operation requirements made by VLSI digital systems makes the design of voltage-mode circuits with high linearity and dynamic range rather difficult. To avoid large voltage swings, a low impedance analog circuit is used with current as the signal instead of the voltage. These can operate at lower power supplies because the current flows through low impedance nodes which in turn will have associated small voltage swings [43]. This factor also impacts on frequency operation because lower impedance in the signal path means a wider frequency band. Furthermore, SI technique does not require linear capacitors thus, they are suitable for standard digital technologies.

Using current mode processing, also allows complex computations to be accomplished with relatively simple circuits using the translinear principle [43]. Originally, translinear circuits were employed with bipolar transistors. However, translinear principle can also be attained with MOS working in subthreshold [44], or even in above threshold [45,46]. The former uses the exponential relationship between the drain current and the gate—source voltage. The latter uses the square law characteristic. A well known example of a circuit that uses this principle is the Gilbert multiplier [47]. An example of a full current mode multiplier using translinear principle was proposed by Wiegerink [48]. Other good areas of applications are in artificial neural networks [7,49] where transcendental equations are often needed.

From the above considerations, there are strong arguments to pursue current as the computational signal environment. Given that a discrete-time approach is preferred for the olfactory model, the next item to be studied is the SI (switch-current) implementation of the K0 second order dynamics.

3.2 The SI Gamma Filter

Switch-capacitor (SC) [39–41] and switch-current (SI) [42,43] circuits are the most common discrete-time techniques used for the design of analog VLSI filters and can be used to implement the K0 dynamics [50]. A survey on SC and SI filters can be found in Appendix A. In this appendix the most common nonideal effects and methods to improve performance are covered and an SC programmable gamma filter, is also presented. However, the same filter implemented with SI circuits, show strong offset dependence with the time constant. In fact, if no type of offset compensation is employed, the final offset may prevent the circuit from operating properly.

Figure 3-1 shows a gamma kernel schematic using SI technique. Transistors M_1, M_2, M_3 and M_4, represent the delay element. Transistor M_5 scales the output by μ (time constant), which is then inverted through M_6 and M_{10} and added to the scaled input current $\mu \cdot i_{k-1}$. The resulting filter function is the gamma kernel defined by Eq. (3-1).

$$i_k(nT) = (1-\mu) \cdot i_k[(n-1)T] + \mu \cdot i_{k-1}[(n-1)T] \qquad (3\text{-}1)$$

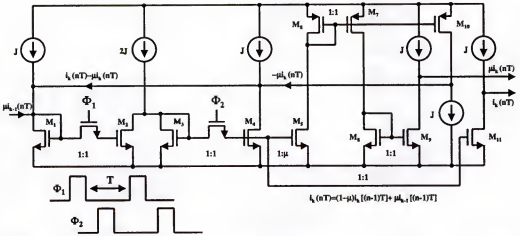


Figure 3-1. The SI gamma kernel (a:b means relative transistor ratios in the loop)

All the methods to overcome nonideal effects inherent to SI circuits presented in the appendix A, can be applied to the gamma filter in Figure 3-1 with minor changes. Since power is a concern maybe even a class AB SI-memory cell could be employed [51–53]. Nonetheless no method is totally effective and always a residual DC offset is present. In most applications this might not be a concern, but for a discrete gamma kernel the residual offset is amplified by the loop gain, and the circuit will not work properly. For a better understanding, assume a first order gamma kernel with a constant value, X_{of}, added at some point of the feedback loop as shown in Figure 3-2a (block diagram representation of Figure 3-1). The X_{of} value represents all open-loop offset contributions measured according to Figure 3-2b. All components in Figure 3-2a are then considered ideal and Eq. (3-2) defines the system difference equation with $X_{k-1}(n) = 0$.

$$X_k(n) = (1-\mu) \cdot X_k(n-1) + X_{of} \tag{3-2}$$

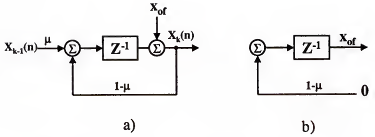


Figure 3-2. Gamma kernel loop with the x_{of} representing all offset contributions throughout the loop

The final steady-state output offset can be determined considering Eq. (3-3), where X_{kof} is a constant and represents the final output value. Substituting Eq. (3-3) into Eq. (3-2) and after a minor manipulation, the final steady-state offset can be defined by Eq.(3-4).

$$[X_k(n) = X_k(n-1)]\Big|_{n \to \infty} = X_{kof} \qquad (3\text{-}3)$$

$$X_{kof} = \frac{X_{of}}{\mu} \qquad (3\text{-}4)$$

According to Eq. (3-4), the closed loop offset is linearly dependent upon the open-loop offset and inversely proportional to the feedback parameter. Minimizing X_{kof} means decreasing X_{of} by using the techniques presented in the literature, but due to the special topology of the gamma kernel, it also requires some extra attention. In fact, with μ less than one, the steady-state offset is an amplified version of the original open-loop offset. Depending on X_{of} and/or μ values, the final offset may become extremely high implying that some form of feedback independent offset cancellation scheme is required. Figure 3-3 shows a Matlab simulation of Figure 3-2a for $X_{of} = 4$ and $\mu = 0.1$ (the simulation limit value matches the result given by Eq. (3-4)). This example shows that an SI circuit with

100μA bias current, would lose almost half of its dynamic range with an accumulated current-loop offset of only 4% (4μA) of the bias current.

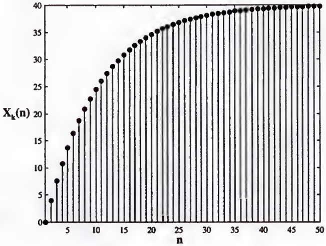


Figure 3-3. Gamma kernel response with zero input and finite offset ($X_{of} = 4$ and $\mu = 0.1$)

A balanced fully differential version of the gamma kernel could, in principle, avoid partially this offset magnification. However, small differential offsets would again be magnified. Furthermore, when chip area needs to be optimized, such as with the olfactory model, silicon area should always be reduced to a minimum and full differential circuits are less optimal in this respect because they need twice the resources and connections of a single-ended system.

3.3 Proposed Feedback Independent Cancellation Scheme for a Gamma Kernel

The method suggested in this section seeks to make the close loop offset equal to the open loop offset, by using a three step procedure. The total offset in the loop is probed, externally memorized, and later subtracted from the signal path. To accomplish cancellation a three phase non-overlapping clock plus a reference clock are needed. Figure 3-4 shows the clock diagram.

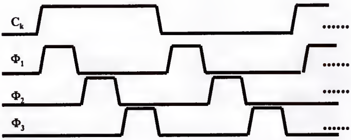


Figure 3-4. New local offset cancellation clock diagram

The input signal must be interleaved with a zero input reference to probe the offset. The zero input reference measures the offset around the loop and is memorized to be later subtracted when the signal input is applied. Both the signal and reference should never overlap otherwise the signal could be subtracted to itself causing errors. Thus, two orthogonal signal paths are created duplicating the delay circuit elements (Figure 3-5). The input signal will flow through one path during phase 1 (Φ_1) and a zero input reference will flow through the other path during phase 2 (Φ_2). At phase 3 (Φ_3) both outputs are updated. Since the clock phases never overlap, the cross-talk between signals is inexistent.

Figure 3-6 helps to illustrate the signal flow at a given clock phase. According to this figure and Figure 3-5, the input is $in(n) = S(n)$ during phase Φ_1, with $S(n)$ the signal to be filtered, and conversely $in(n) = 0$ during phase Φ_2. At phase Φ_3 the output delay elements D_{11}, and D_{22} are simultaneously updated. Consider that *Ck="on"* and that the phase is Φ_2 $(in(n) = 0)$. According to Figure 3-5, and the timing diagram of Figure 3-6, the output of D_{22} will be the accumulated offset around the loop called *Ref(n)*. This reference signal may now be memorized externally (offset subtraction block in Figure 3-5) and used to subtract the output offset from the input signal flow path for offset cancellation.

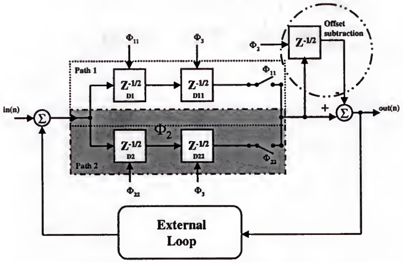


Figure 3-5. Offset cancellation concept

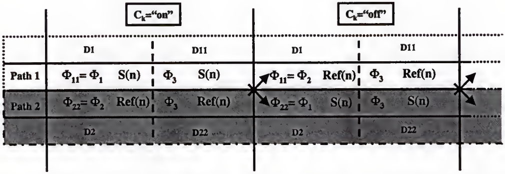


Figure 3-6. Timing diagram to illustrate what signal is present in each delay element of Figure 3-5

Tracking the right offset (e.g., there is no guarantee that the offset in both the signal and reference paths are equal) still remains a problem. To guarantee that the reference signal, *Ref(n),* matches the signal path offset, the role of the signal and the offset circuit paths are interchanged in each half period of clock C_k (Figure 3-5 and Figure 3-6). The external memory (offset subtraction block in Figure 3-5) will be holding the correct offset needed for compensation anticipating in this manner the next phase offset value in the signal path. It should be noted that the offset is subtracted both to the signal and offset paths, such that the reference signal is only the open-loop accumulated offset and not the integrated value given by Eq. (3-4). The desired clock switching necessary to interchange the paths roles on each clock-cycle can easily be accomplished with simple logic gates as shown in Figure 3-7.

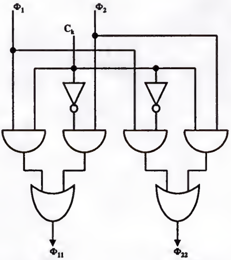


Figure 3-7. Logic to generate Φ_{11} and Φ_{22} clocks

Figure 3-8 shows a schematic for simulation in Hspice [54]. The feedback (external loop) was implemented with the help of a functional current source with an added offset of 4μA. All transistors have sizes $W/L = 50\mu/12\mu$ and $J = 50\mu A$. The model in all simulations is 1.2μm Level 3 for 1.2μm AMI—MOSIS technology. The Hspice CAPOP option is set to 9 for charge conservation calculations. All source currents are transistors.

Figure 3-9 and Figure 3-10 show the simulation outcome for two different μ values (0.2 and 0.02 respectively). In Figure 3-9 part (1), no cancellation is applied and the output goes up to -15 μA, but as compensation is turned "on" (part (2)) the offset quickly approaches 1.1 μA. When a square pulse is applied (part (3)) the system integrates the input and the final offset remains at 1.1 μA. Hence, there is no interference between the offset cancellation and the signal itself. However, it is required that the input to be sampled at Φ_1 with a return to zero sampling.

Figure 3-10 presents the simulation result for μ=0.02 showing that with small μ values the offset can be extremely high (48 μA). But as soon as the compensation circuit is activated the offset drops dramatically to -4μA. The offset compensation still exhibits dependence over the μ parameter, but much less than without compensation.

The simulation results show that the technique avoids largely the offset dependence on the feedback parameter. In the following sections, theoretical support will be presented to help understand better the technique.

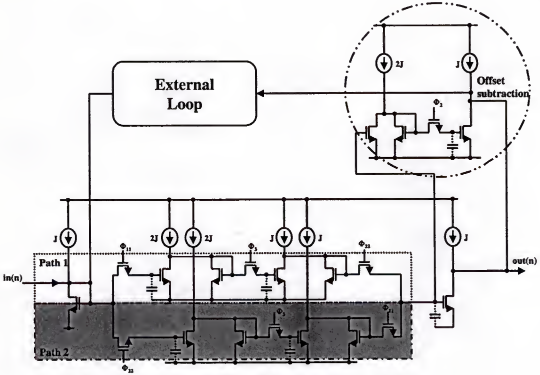


Figure 3-8. Transistor level representation of Figure 3-5

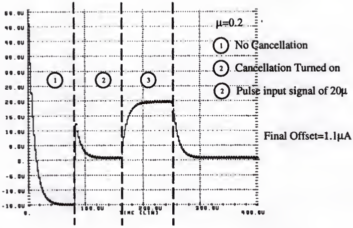


Figure 3-9. Simulation results μ=0.2. 1) No cancellation; 2) Cancellation turned on; 3) Square input signal of 20mA

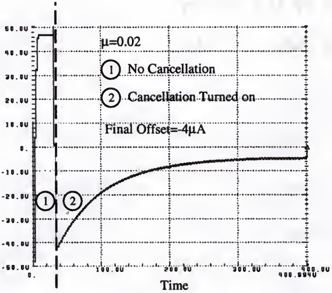


Figure 3-10. Simulation results μ=0.02. 1) No cancellation; 2) Cancellation turned on

3.4 Proof of Convergence

Before going into experimental results it is necessary to investigate under what conditions the offset compensation converges. To proceed with a proof, a more formal representation of the method is required. Such a formal representation is depicted in Figure 3-11.

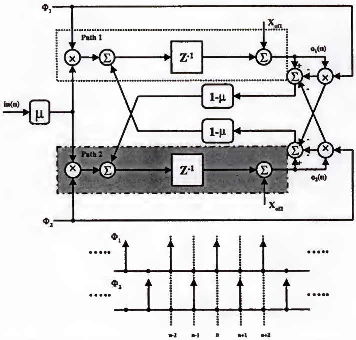


Figure 3-11. Local offset compensation formal representation

In Figure 3-11, the common feedback block $1-\mu$ of Figure 3-5 is broken into two equal feedback blocks to allow the update of the input signal and offset reference signal at the same time. The separation is controlled by the signals Φ_1 and Φ_2 together with the associated multipliers. Given that $\Phi_2(n) = \Phi_1(n-1)$ and assuming $\Phi_1(0) = \delta(n)$, according to the signal representation in Figure 3-11, the input signal $i(n)$ will flow through path 1 when n is even and through path 2 when n is odd. Conversely the zero reference signal will be flowing through the same paths in opposite phases. Since the feedback $1-\mu$ is crossed at the output, the signal and reference will be crossed. Finally, the output multipliers and adders represent the offset subtraction block. Note that when n is even, the input to the delay block in path 1 is the signal but its delayed output is the reference. This reference is subtracted to the output of path 2 which represents the delayed signal to be scaled and added to $i(n)$. This very same reference is also subtracted to itself, scaled and added to the zero reference signal input in path 1. The loop offsets, Xof1 and Xof2, are measured according to Figure 3-2, and the single ended filter output $O(n)$ is obtained by crossing the outputs signals $O_1(n)$, and $O_2(n)$ as represented in Figure 3-12.

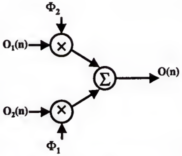


Figure 3-12. Crossing signals Φ_1 and Φ_2, at the outputs of Figure 3-11 to obtain the filter signal output

Formal representations for the signals Φ_1 and Φ_2 can be formulated as Eq. (3-5). This description of Figure 3-11 matches the same technique described for Figure 3-5 but with a more suitable representation for mathematical manipulations

$$\begin{aligned}\Phi_1(n) &= \sum_{k=0}^{\infty} \delta[n-2\cdot k] \\ \Phi_2(n) &= \Phi_1(n-1) = \sum_{k=0}^{\infty} \delta[n-(2\cdot k+1)]\end{aligned} \tag{3-5}$$

The outputs $O_1(n)$ and $O_2(n)$ in Figure 3-11 can then be formulated as follows (without loss of generality consider $in(n) = 0$):

$$\begin{aligned} O_1(n) &= (1-\mu)\cdot\{[1-\Phi_2(n-1)]\cdot O_2(n-1)-\Phi_1(n-1)\cdot O_1(n-1)\}+ \\ &\qquad X_{of1} \\ O_2(n) &= (1-\mu)\cdot\{[1-\Phi_1(n-1)]\cdot O_1(n-1)-\Phi_2(n-1)\cdot O_2(n-1)\}+ \\ &\qquad X_{of2} \end{aligned} \tag{3-6}$$

Considering both Eq. (3-5) and Eq. (3-6), the output signals can be reformulated in the form of Eq. (3-7) and Eq. (3-8).

$$O_1(n) = \begin{cases} X_{of1} & ,n\ even \\ (1-\mu)\cdot[O_2(n-1)-O_1(n-1)]+X_{of1} & ,n\ odd \end{cases} \tag{3-7}$$

$$O_2(n) = \begin{cases} (1-\mu)\cdot[O_1(n-1)-O_2(n-1)]+X_{of2} & ,n\ even \\ X_{of2} & ,n\ odd \end{cases} \tag{3-8}$$

Before analyzing convergence further, lets find out what offset in Figure 3-11 would be present if no cancellation procedure were applied. Equivalently if the signals represented by Φ_1 and Φ_2, at the output multipliers, were zero. In this situation Eq. (3-6) could be written as Eq. (3-9).

$$\begin{aligned} O_1(n) &= (1-\mu)\cdot O_2(n-1)+X_{of1} \\ O_2(n) &= (1-\mu)\cdot O_1(n-1)+X_{of2} \end{aligned} \tag{3-9}$$

To find the offset when $n\to\infty$ then $O_1(n) = O_1(n-1) = X_{ofT1}$ and $O_2(n) = O_2(n-1) = X_{ofT2}$. Taking this equalities, substituting in Eq. (3-9) and solving the system of equations results in Eq. (3-10).

$$\begin{aligned} X_{ofT1} &= \frac{1}{\mu\cdot(2-\mu)}\cdot[(1-\mu)\cdot X_{of2}+X_{of1}] \\ X_{ofT2} &= \frac{1}{\mu\cdot(2-\mu)}\cdot[(1-\mu)\cdot X_{of1}+X_{of2}] \end{aligned} \tag{3-10}$$

As expected, when $\mu \rightarrow 0$ both offsets tend to infinity. Figure 3-13 shows the simulation result of Figure 3-11 using Matlab.

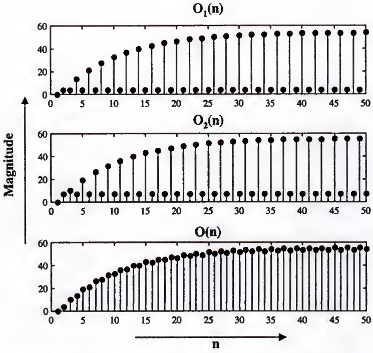


Figure 3-13. Figure 3-11 simulation result without offset subtraction
$(\mu = 0.1, X_{of1} = 4, X_{of2} = 7)$

The two signal paths alone do not improve offset. The final (large time) offsets measured from Matlab simulation were $X_{ofT1} = 54.21, X_{ofT2} = 55.79$, which matched the values predicted by Eq. (3-10) for the simulation values (Figure 3-13).

Consider now the offset subtractor activated and assume initial conditions $O_1(n-1) = X_1$ and $O_2(n-1) = X_2$. Iterating equations (3-7) and (3-8) the close form can be proven to be equations (3-11) and (3-12) (Appendix B).

$$n = even \rightarrow \begin{cases} O_1(n) = X_{of1} \\ O_2(n) = (1-\mu)^{n+1} \cdot (X_1 - X_2) + \left(\sum\limits_{k=0}^{n} [(1-\mu)^k \cdot (-1)^k] \right) \cdot X_{of2} + \\ \left(-\sum\limits_{k=0}^{n} [(1-\mu)^k \cdot (-1)^k] + 1 \right) \cdot X_{of1} \end{cases} \tag{3-11}$$

$$n = odd \rightarrow \begin{cases} O_1(n) = (1-\mu)^{n+1} \cdot (X_1 - X_2) + \left(-\sum\limits_{k=0}^{n} [(1-\mu)^k \cdot (-1)^k] + 1 \right) \cdot X_{of2} + \\ \left(\sum\limits_{k=0}^{n} [(1-\mu)^k \cdot (-1)^k] \right) \cdot X_{of1} \\ O_2(n) = X_{of2} \end{cases} \tag{3-12}$$

$$\begin{aligned} n = even \rightarrow & \begin{cases} O_1(n) = X_{of1} \\ X_{ofT2} = O_2(n) = \frac{1}{2-\mu} \cdot X_{of2} + \frac{1-\mu}{2-\mu} \cdot X_{of1} \end{cases} \\ n = odd \rightarrow & \begin{cases} X_{ofT1} = O_1(n) = \frac{1-\mu}{2-\mu} \cdot X_{of2} + \frac{1}{2-\mu} \cdot X_{of1} \\ O_2(n) = X_{of2} \end{cases} \end{aligned} \tag{3-13}$$

To find out the asymptotic output, the limit when $n \rightarrow \infty$ has to be calculated. Taking into consideration that the summation in brackets represent the summing terms of a geometric progression with ratio $r = -(1-\mu)$ and that the stability guarantees always $(1-\mu) < 1$, then the limits are defined by Eq. (3-13). Considering $0 < \mu < 1$ and assuming $X_{of1} < X_{of2}$ then:

$$
\begin{aligned}
n = even \rightarrow & \begin{cases} O_1(n) = X_{of1} \\ \left(\frac{1}{2}\cdot(X_{of2}+X_{of1})\right) < X_{ofT2} < X_{of2} \end{cases} \\
n = odd \rightarrow & \begin{cases} X_{of1} < X_{ofT1} < \left(\frac{1}{2}\cdot(X_{of2}+X_{of1})\right) \\ O_2(n) = X_{of2} \end{cases}
\end{aligned}
\qquad (3\text{-}14)
$$

Equation (3-13) shows that the offset cancellation scheme converges to finite values. It is still dependent on the feedback parameter but if the filter is low-pass, the limits in Eq. (3-14) show that the variation over the parameter is negligible. Actually if both offsets match there is no dependence over the feedback parameter. In reality, the final value is upperbounded by the biggest accumulated offset measured as in Figure 3-2. It is either that value or smaller. If however $1 < \mu < 2$, the offset will be increasing. For these values of μ the filter is high-pass and it is unreasonable to use the offset compensation because as Eq. (3-4) states, the offset is attenuated if $\mu > 1$. We may conclude that the offset cancellation scheme should be used only when $\mu < 1$. In fact this is the most frequent occurrence with signals that have a *1/f* characteristic and for which the gamma filter is better fitted [31].

Figure 3-14 shows a simulation result with offset subtraction. As predicted from earlier comments and equations, the offset cancellation method tends to level-out the two path offsets. The final values measured with Matlab are $X_{ofT1} = 5.421$ and $X_{ofT2} = 5.579$ which match the values calculated with Eq. (3-13).

Figure 3-15 represents the same simulation but with nonzero initial conditions and input. The result equals the ideal gamma filter shifted by the same offset amount. Equation (3-13) is plotted in Figure 3-16 for different values of μ. As expected, decreasing μ will attenuate the individual open-loop offsets, X_{of1} and X_{of2}, but will also make the final offsets close to each other. Nevertheless, Figure 3-16 tells that if the open-loop offsets are unmatched, the output signal $O(n)$ will be affected with a summing signal that results from the compensation procedure. Yet, the signal is a sinusoid of frequency $f_s/2$ with magnitude and DC values that can be calculated from Eq. (3-13) and can be easily removed[1]. For the same example of Figure 3-14 and taking the FFT for the region far from the transient, Figure 3-17 is obtained where the magnitude values can be calculated as follows:

$$\begin{cases} A \cdot \sin(2 \times \pi \times f_d) + DC \\ A = (X_{ofT2} - X_{ofT1})/2 = 0.0789 \\ DC = (X_{ofT2} + X_{ofT1})/2 = 5.5 \end{cases} \tag{3-15}$$

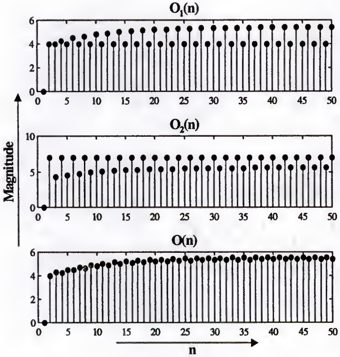


Figure 3-14. Figure 3-11 simulation result with offset subtraction ($\mu = 0.1, X_{of1} = 4, X_{of2} = 7$.

[1] In fact this signal would be automatically removed with the smoothing filter.

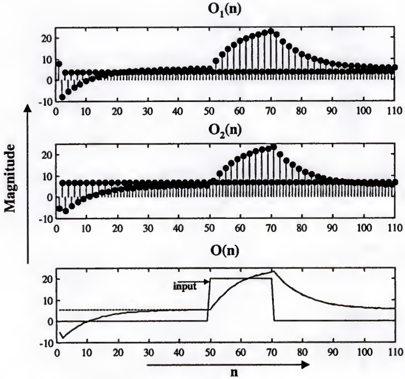


Figure 3-15. The same as Figure 3-14, but with input signal and initial conditions $O_1(n-1) = 8, O_2(n-1) = -5$. Dashed line corresponds to an ideal gamma filter with the output added with a constant equal to the final offset of $O(n)$

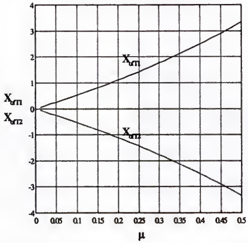


Figure 3-16. Final offsets as functions of feedback parameter μ. (X_{of1}=10; X_{of2}=-10)

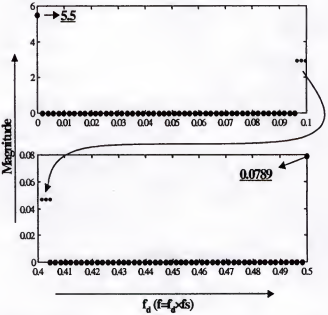


Figure 3-17. The FFT of $O(n)$ in Figure 3-14, for a region far from the initial transitory response

3.5 Other Nonideal Effects that may Affect the Proposed SI Offset Cancellation Method

To take into account possible gain mismatches in a real implementation, gain factors are added into different points of the offset model as illustrated in Figure 3-18.

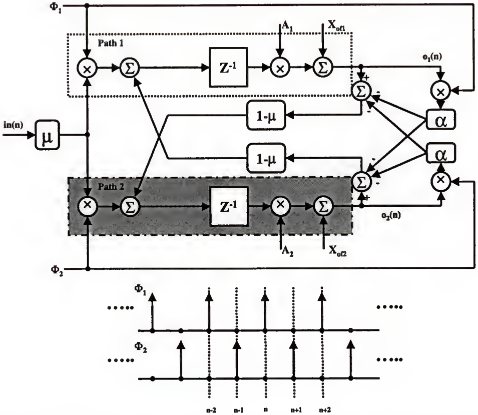


Figure 3-18. Offset cancellation formal representation with gain errors associated

The variables A_1 and A_2, represent the gain errors associated with the different delays and α represents the gain error associated with the offset subtraction block (Figure 3-5). With the added terms, equations (3-7) and (3-8) can be rewritten as Eq. (3-16) and Eq. (3-17).

$$O_1(n) = \begin{cases} A_1(1-\mu)(1-\alpha)\cdot O_2(n-1)+X_{of1} & ,n\ even \\ A_1(1-\mu)\cdot[O_2(n-1)-\alpha O_1(n-1)]+X_{of1} & ,n\ odd \end{cases} \qquad (3\text{-}16)$$

$$O_2(n) = \begin{cases} A_2(1-\mu)\cdot[O_1(n-1)-\alpha O_2(n-1)]+X_{of2} & ,n\ \ even \\ A_2(1-\mu)(1-\alpha)\cdot O_1(n-1)+X_{of2} & ,n\ \ odd \end{cases} \qquad (3\text{-}17)$$

Comparing these equations with the former set (equations (3-7) and (3-8)), the main change is in the reference signal phase, which occurs for $O_1(n)$ when n is even, and for $O_2(n)$ when n is odd. It is no longer the accumulated open-loop offset. The effect on the reference signal as $n \to \infty$ can be calculated realizing that during the reference signal phase $O_1(n) = O_1(n-1) = X_{ofT1}$ and $O_2(n) = O_2(n-1) = X_{ofT2}$.

Assume at this point that the offset cancellation is connected during the reference signal phase and not connected during the signal phase for $-\infty < n < 0$. The connection for the signal phase is applied at $n = 0$. In this case the new reference signal is stable at $n = 0$ and equations (3-16) and (3-17) can be written as equations (3-18), (3-19), and (3-20)

$$O_1(n) = \begin{cases} X_{ofT1} & ,n\ \ even \\ A_1(1-\mu)\cdot[O_2(n-1)-\alpha O_1(n-1)]+X_{of1} & ,n\ \ odd \end{cases} \qquad (3\text{-}18)$$

$$O_2(n) = \begin{cases} A_2(1-\mu)\cdot[O_1(n-1)-\alpha O_2(n-1)]+X_{of2} & ,n\ \ even \\ X_{ofT2} & ,n\ \ odd \end{cases} \qquad (3\text{-}19)$$

$$\begin{cases} X_{ofT1} = \dfrac{A_1(1-\alpha)(1-\mu)}{1-A_1A_2(1-\alpha)^2(1-\mu)^2}\cdot X_{of2} + \dfrac{1}{1-A_1A_2(1-\alpha)^2(1-\mu)^2}\cdot X_{of1} \\ X_{ofT2} = \dfrac{A_2(1-\alpha)(1-\mu)}{1-A_1A_2(1-\alpha)^2(1-\mu)^2}\cdot X_{of1} + \dfrac{1}{1-A_1A_2(1-\alpha)^2(1-\mu)^2}\cdot X_{of2} \end{cases} \qquad (3\text{-}20)$$

The worst case scenario happens when $\mu \to 0$. Taking the limit of (3-20) results in (3-21).

$$\begin{cases} X_{ofT1} = \dfrac{A_1(1-\alpha)}{1-A_1A_2(1-\alpha)^2}\cdot X_{of2} + \dfrac{1}{1-A_1A_2(1-\alpha)^2}\cdot X_{of1} \\ X_{ofT2} = \dfrac{A_2\cdot(1-\alpha)}{1-A_1A_2(1-\alpha)^2}\cdot X_{of1} + \dfrac{1}{1-A_1A_2(1-\alpha)^2}\cdot X_{of2} \end{cases} \qquad (3\text{-}21)$$

Forcing a 20% error in all gains ($A_1 = 1.2, A_2 = 0.8, \alpha = 0.8$), which is not unreasonable in a real situation, Eq. (3-21) evaluates to:

$$\begin{cases} X_{ofT1} \cong \dfrac{X_{of2}}{4} + 1.04\cdot X_{of1} \\ X_{ofT2} \cong \dfrac{X_{of1}}{6} + 1.04\cdot X_{of2} \end{cases} \qquad (3\text{-}22)$$

The result shows that even with relatively imprecise current mirrors the increase in the reference signal from the ideal X_{of1} and X_{of2} is comparable to the current mirror imprecision. In this case the biggest increase is about 30% in both X_{ofT1} and X_{ofT2} (considering equal open-loop offsets), which is not significant considering that we are at the limit case where $\mu = 0$. In reality $0 < \mu < 1$ which will make X_{of1} and X_{of2} close to the ideal values. With a careful design of the offset subtraction block to make α close to one will greatly improve the performance. Without changing any of the previous parameters except increasing α to 0.9 will reduce the offset to about a 15% increase.

It should be noted that the gain imprecision affects the loop gain. For this reason the inequality given by Eq. (3-23) has to be always guaranteed otherwise the filter goes unstable.

$$A_1A_2(1-\alpha)^2(1-\mu)^2 < 1 \qquad (3\text{-}23)$$

The proof of convergence for equations (3-18) and (3-19) can be found in appendix C. The proof was performed using the approximation for the reference phase defined in Eq. (3-24). The final result is transcribed in Eq. (3-25).

$$\begin{cases} X_{ofT1} \cong X_{of1} \\ X_{ofT2} \cong X_{of2} \end{cases} \tag{3-24}$$

$$\begin{aligned} (n = odd) \rightarrow & \begin{cases} O_1(n) = \left((1-\mu)A_1 \dfrac{1-\alpha(1-\mu)A_2}{1-(1-\mu)^2 A_1 A_2}\right) \cdot X_{of2} + \\ \left(\dfrac{1-\alpha(1-\mu)A_1}{1-(1-\mu)^2 A_1 A_2}\right) \cdot X_{of1} \\ O_2(n) \cong X_{of2} \end{cases} \\ (n = even) \rightarrow & \begin{cases} O_1(n) \cong X_{of1} \\ O_2(n) = \left(\dfrac{1-\alpha(1-\mu)A_2}{1-(1-\mu)^2 A_1 A_2}\right) \cdot X_{of2} + \\ \left((1-\mu)A_2 \dfrac{1-\alpha(1-\mu)A_1}{1-(1-\mu)^2 A_1 A_2}\right) \cdot X_{of1} \end{cases} \end{aligned} \tag{3-25}$$

As already mentioned, the offset cancellation is effective if $\mu < 1$. For this range of values, if the error associated with α is in the same range as A_1 and A_2 and if Eq. (3-26) is true (which is a condition to be met otherwise the filter goes unstable) it is guaranteed that the constants affecting X_{of1} and X_{of2} in Eq. (3-25) are less than one.

$$(A_1(1-\mu) < 1) \wedge (A_2(1-\mu) < 1) \tag{3-26}$$

3.6 A Real Chip Test for the SI Integrator Offset Compensation

To test the offset compensation methodology, a gamma filter with programmable feedback parameter μ was designed with 1.2μm AMI–MOSIS technology. The complete schematic of the designed cell is shown in Figure 3-19.

All transistors are $W/L = 25.2\mu m/6\mu m$ except the ones at the current mirror within the feedback block that were set to values such that $\mu_1 \cong 0.14$ and $\mu_2 \cong 0.57$. The parameter is selected actuating at the transistors μ_1 and μ_2. A switch is added to control externally the offset compensation connection state (offset on/off transistor). The bias currents were set to 50μA. This low bias current is intentionally set to increase the offset due to clock feedthrough [55–57] (if needed, an introduction to CFT can be found in Appendix A), and actually verify that the signal independent offset is cancelled.

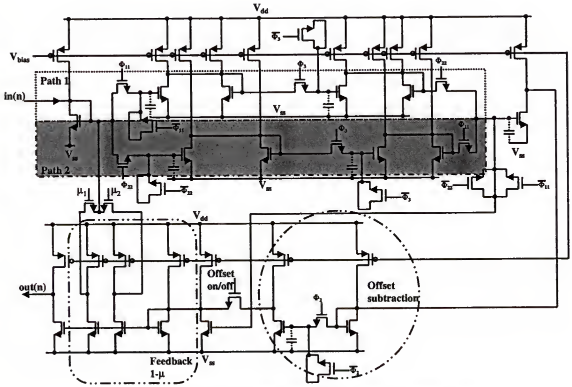


Figure 3-19. The SI gamma filter with offset compensation.

Figure 3-20 shows the measured transient result for the two implemented gains. In the graph, the signal and reference phase updates are plotted together. The circuit was allowed to run without offset compensation for some time. After the output signal settles, the compensation is set to "on". This moment is represented by the vertical line in Figure 3-20. As expected, the reference signal is set to the final offset value on the first clock cycle, while the signal phase needs some time to converge to a final value. This was predicted by the theory and is due to the orthogonal characteristics of the reference and signal phases. The convergence time is longer for smaller values of μ, which is coherent with the result of Eq. (3-12), and after some time the offsets tend to equalize. This is well represented in Figure 3-21, where both steady state offsets are compared.

The results clearly show different offset values for each path and that the final values are strongly dependent on the feedback when the compensation is "off". Without compensation, this filter topology would be worthless with offsets close to the total bias current. However with compensation, the offset in the different paths tend to equalize and are decreased to reasonable values. Table 3-1 shows that the final offset values are slightly dependent on the feedback parameter, but not as much as without compensation. Finally, it should be noted that the output transistors are not in the loop, which means that the measures presented so far are a contribution of the offset from this cell plus the offset of the overall filter. From the results, it can then be concluded that the compensation technique is feasible and that the theory developed is a good description of the measurements.

3.7 Harmonic Distortion and Improvements

Harmonic distortion is another effect caused by the gain mismatches discussed in Section 3.5. In fact if the gain on both paths is not equal, the input signal is commuted between two different systems at every other sampling time. Figure 3-22 shows two plots demonstrating the resulting harmonic distortion.

The equivalent gamma filter provides the same response as Figure 3-22 but misses the second line around $0.44 \times f_s$ thus, this line represents harmonic distortion. However it gets smaller as the gain mismatches decrease. If the gain is made equal on both paths, the harmonic distortion completely disappears.

Table 3-1.Measured steady state offset maximum values

Feedback/ Compensation	$\mu_1 \cong 0.14$ (μA)	$\mu_2 \cong 0.57$ (μA)
'off'	**41.6**	**28.8**
'on'	**7.6**	**8.4**

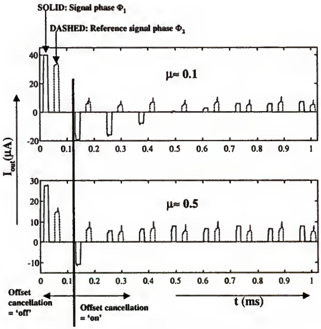


Figure 3-20. Measured offset cancellation transient. The moment where the compensation is set to "on" is marked by a vertical line in the graph

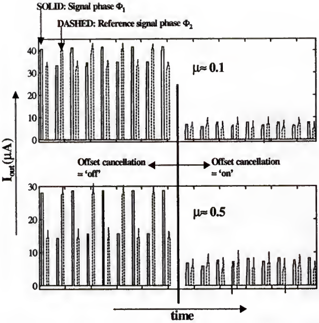


Figure 3-21. Measured steady state offsets before and after offset compensation connection

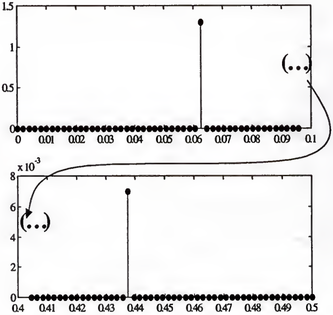


Figure 3-22. Gamma filter response with offset cancellation. The gains are affected by 1% error

An equivalent procedure to Figure 3-5 can be employed rendering no harmonic distortion due to gain mismatch. The new clocking scheme is shown in Figure 3-23 and the new method is depicted in Figure 3-24.

In operational terms, the new offset compensation schematic is very similar to Figure 3-5, but needs only a single path for the signal and reference phases. The delay D2 is the intermediate memory needed to keep track of the previous signal update. The clock scheme of Figure 3-23 guarantees that the correct delayed signal is present at D3 when the input D1 is updated. Equations (3-17) and (3-18) are perfectly valid for the present situation[1] with the advantage that $A_1 = A_2$ and $X_{of1} = X_{of2}$, then no harmonic distortion or offset oscillation results. Another advantage is that it uses one less memory cell than the previous method. Yet, comparing with Figure 3-4 and Figure 3-5, the clocking scheme is more complex and imposes lower maximum frequency operation due to the extra phases between each input sampling time Φ_1.

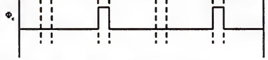


Figure 3-23. New offset compensation clock scheme.

[1] Actually all the techniques and results found in the previous sections are directly applied to the present method

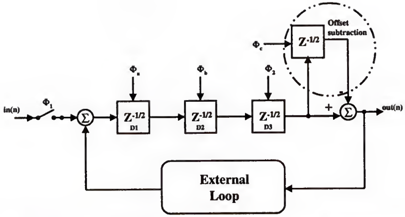


Figure 3-24. Representation of the new offset cancellation concept

3.8 Summary

This chapter has presented a new local single-ended offset compensation method using SI circuits, to reduce the offset amplification verified with gamma kernels when $\mu < 1$. It was mathematically shown under what conditions the system converges, studies on nonideal effects were presented, and conclusions drawn concerning the minimization of those effects. Experimental measures also confirmed theoretical results.

CHAPTER 4
FILTER AND HOLD (F&H)—A NOVEL DISCRETE-TIME FILTERING TECHNIQUE

4.1 Introduction

Switch-capacitor and switch-current are the preferred techniques for discrete-time processing. The former can accomplish very high precision time constants and the last is suitable for standard technologies. Nevertheless, for big systems with repeated similar blocks, neither technique is area efficient and the power cost does not comply with the nano-power requirements. Low-area and nano-power consumptions are usually correlated. Simpler circuits take less elements and in general will render lower power. This chapter presents a novel mixed time technique that, in essence, is a continuous-time system, but uses integration switching to get very long time constants with little area and, if needed, low power. The technique was denominated "Filter and Hold (F&H)" [58] and this chapter presents a full mathematical framework to design arbitrary F&H filters. The procedure of using switching to increase a resistor value was outlined by Lancaster [59] in 1975, but the author of this dissertation was unware of its existence until recently.

The novel F&H sampled data technique bases its functional principle on time gating as a multiplication factor to scale time constants. In practice the effect can be observed as a capacitor multiplication because in a practical active filter any time constant is associated with a capacitor. The multiplication is achieved by periodically holding the capacitor voltage (or halting the integration process) after a brief period of charge (discharge). are therefore achieved with F&H designs. There are various instances where long-time constants or low frequency filters are needed. Typically for 60Hz or 50Hz cancellation, biological signal conditioning and modeling, or audio front-end applications. Typical continuous-time Gm-C VLSI filters have lower end cut-off frequencies in the order of magnitude of tens of kHz, which is clearly insufficient for those applications that require frequency ranges in the order of tens and hundred of Hz. The F&H technique is then well fitted for the design of the olfactory model.

In more specific terms, the F&H technique is a mixed analog/discrete-time design that combines analog continuous-time conventional filtering with sampling. The technique proposed here is based on the very intuitive idea of permitting a capacitor to integrate a current during τ seconds and halting it during a time of $T - \tau$ seconds. This process is repeated every T seconds. The resulting time constant value is then multiplied by the duty-cycle, which is defined as the ratio of τ over the sampling period T. The multiplicative effect to the time constant is solely based on the duty-cycle of the clock signal and the sampling period. Therefore, large time constants can be achieved without a decrease in sampling frequency or with high capacitor ratios (or transistors in SI systems). The switching scheme is simple and both the area and power requirements can be made small.

4.2 F&H Concept

The F&H concept can be easily explained for 1st order high-pass (Figure 4-1) and low-pass filters (Figure 4-5). Consider the inputs and outputs sampled and hold, yielding the discrete-time F&H. Although the F&H is a discrete-time system in terms of input/output, in reality it behaves as a hybrid continuous-time/discrete-time system.

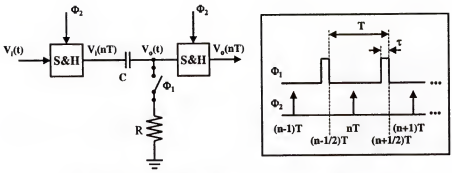


Figure 4-1. High-pass F&H, and switching clock

Take the high-pass F&H example of Figure 4-1. Current is allowed to flow through the capacitor during a brief period of time τ. For the rest of the time period, $T - \tau$, this current is zeroed. Interrupting the current suddenly does not violate physical principles since the capacitor voltage is an integral of this current, hence it does not suffer abrupt changes.

When the clock Φ_1 turns to "on" state at $t = (n - 1/2) \cdot T - \tau$, the input voltage is already fixed to a value $V_i[(n-1) \cdot T]$. If the capacitor has an initial voltage different from zero, $V_c = V_i[(n-2) \cdot T] - V_o[(n-1) \cdot T]$, then the difference equation of the system can be found by the step response resulting in Eq. (4-1). The respective Z transform can be defined by (4-2) where $k = \tau/T$ represents the duty-cycle of Φ_1.

$$V_o(nT) = (V_o[(n-1)T] + \{V_i[(n-1)T] - V_i[(n-2)T]\}) \cdot e^{-\tau/(RC)} \qquad (4\text{-}1)$$

$$\frac{V_o(z)}{V_i(z)} = e^{-\tau/(RC)} \cdot \frac{1 - z^{-1}}{1 - e^{-\tau/(RC)} \cdot z^{-1}} \cdot z^{-1}$$

$$\Leftrightarrow \qquad (4\text{-}2)$$

$$\frac{V_o(z)}{V_i(z)} = e^{-kT/(RC)} \cdot \frac{1 - z^{-1}}{1 - e^{-kT/(RC)} \cdot z^{-1}} \cdot z^{-1}$$

The filter represented by Eq. (4-2) is indeed an high-pass filter with a zero at $z = 1$, and a pole at $z = e^{-kT/(RC)}$. Depending on the value of k, this pole can be made close to $z = 1$, without changing the time period T nor C. However, keeping k constant, changing T does not change the pole position relative to the signal bandwidth. The sampling period changes both input sampling and the system bandwidths. The relative decrease or increase of pole position is followed by the same amount of relative input bandwidth compression or expansion in terms of the normalized frequency $|W| = |T \cdot \Omega| < \pi$. This is not the case for a SI or SC circuits. The pole and zero positions depend only on transistor and capacitor ratios, respectively. Consequently a different sampling frequency will change only the relative signal bandwidth in the normalized frequency spectrum but the filter pole and zero positions are kept unchanged. Thus, in opposition to the F&H case, changing the sampling period in an SC filter affects the filtered signal. Varying the sampling period affects the signal time resolution. The same happens with F&H but in this case the filtering characteristic is kept unchanged. To better illustrate the above explanation, a simulation was performed over the SC low-pass gamma filter shown in Figure A-4 (Appendix A). Two different sampling frequencies were used. The simulation result is shown in Figure 4-2. It is clear that for the same μ value, defined by the capacitor ratio in Eq. (A-9) (Appendix A), the behavior depends on the sampling period (the input is kept unchanged).

Figure 4-3 shows a simulation result for the F&H in Figure 4-1 together with the continuous-time prototype response. The resistor and capacitor were set to $R = 10k\Omega$ and $C = 1nF$ respectively and the duty-cycle is k=10%. The capacitor for the continuous-time prototype is set to $C = 10nF$. As expected, the response of the F&H is the same as the RC continuous-time filter except that the output is sampled. In Figure 4-4 a zoom in of a squared area is shown for better understanding. With the same duty-cycle, the simulation shows that the filter behavior is not affected by the sampling period, only the time resolution changes. Another interesting aspect, and the actual main useful practical feature, is parameter scaling. As mentioned earlier, the time constant is scaled by k. Given that k is less than one, a $k \cdot C$ smaller capacitor can be used to achieve an equivalent $\tau = R \cdot C$ time constant. Theoretically the capacitor can be made as small as desired, however it is not difficult to attain realizable physical limitations.

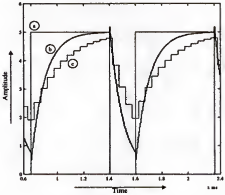


Figure 4-2. SC gamma filter with $\mu = 0.2$. a) Input; b) Output with $T = 5\mu s$; c) Output with $T = 50\mu s$

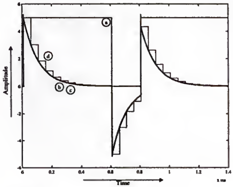


Figure 4-3. Two high-pass F&H results (duty-cycle k=10% in all) over the same input signal. a) Input; b) Output of Figure 4-1 with the switch always on (continuous-time equivalent) but with capacitor 10 times bigger; c) Output with $T = 5\mu s$; d) Output with $T = 50\mu s$

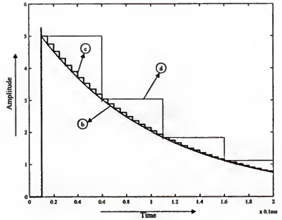


Figure 4-4. Zoom over Figure 4-3 first 200μs (the indexes represent the same as the ones in Figure 4-3)

The same procedure can be applied to a low-pass filter configuration. As with the high-pass filter, a switch is strategically inserted to periodically switch the current flowing through the capacitor as shown in Figure 4-5

Applying to this case the same method used to find the high-pass F&H result, the low-pass F&H transfer function can be found to be Eq. (4-3), and the same facets discussed earlier for the high-pass F&H apply here.

$$\frac{V_o(z)}{V_i(z)} = \frac{1-e^{-kT/(RC)}}{1-e^{-kT/(RC)}\cdot z^{-1}}\cdot z^{-1} \qquad k = \frac{\tau}{T} \tag{4-3}$$

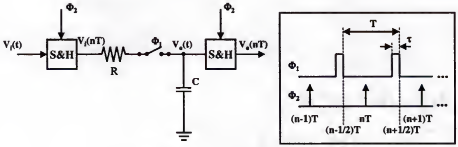


Figure 4-5. Low-pass F&H

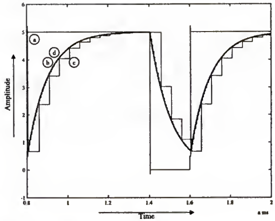


Figure 4-6. Two low-pass F&H results (duty-cycle k=10% in all) over the same input signal. a) Input; b) Output with $T = 5\mu s$; c) Output with $T = 50\mu s$; d) Output of Figure 4-1 with the switch always on (continuous-time equivalent) but with capacitor 10 times bigger

Band-pass and band-reject filters can be assembled from the F&H high-pass and low-pass sections. In this perspective, the first order F&H is general and can be applied for the four known frequency filtering characteristics. However, can other filter topologies be implemented with a F&H configuration? In what conditions can the technique be applied? In summary how general is the F&H technique? This is an important question that needs to be addressed in order to attest the importance and generality of the technique and will be the theme of the following sections.

4.3 Generality of the F&H Discrete-Time Filter

To prove the F&H generality, two systems will be considered. The systems are similar with a difference, one has the dynamic matrices scaled while the other has the states halted periodically by a clock signal displaying a duty-cycle: $k = \tau/T$. The input signal is sampled on a different phase (as done with the first order sections in Figure 4-1 and Figure 4-5). This sampled and hold signal is applied to both systems, and if they exhibit the same output for every $n \cdot T$ seconds with $n \in \aleph$, then from an input/output perspective the systems are equivalent. For simplicity the initial conditions are considered to be zero. Figure 4-7 illustrates graphically what is the expected output behavior for the two systems (continuous-time versus F&H).

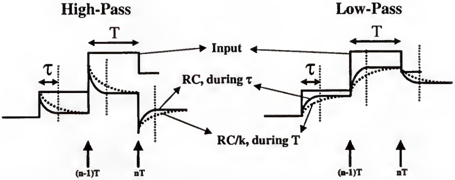


Figure 4-7. Graphical representation of a generic F&H response (solid) compared with the continuous-time filter (dashed) with RC/k slower time constant over the same sample and hold input

Consider an input signal defined as Eq. (4-4), where • denotes convolution operation and *rect*(.) is the pulse function defined in Eq. (4-5). For better understanding, Figure 4-8 shows a graphical representation of these mathematical operations.

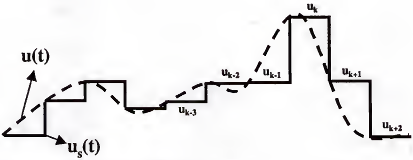


Figure 4-8. Sample and hold version of a continuous-time signal *u(t)*

Consider now a generic realization of a linear system described by a set of state space equations (4-6) [60]. Equation (4-6) already explicitly portrays the sampled and hold input signal $u_s(t)$ defined in Figure 4-8.

$$\begin{aligned} u_s(t) &= u(t) \cdot \left(\sum_{k=0}^{\infty} \delta(t-kT) \right) \bullet rect\left(\frac{t-T/2}{T}\right) \Leftrightarrow \\ u_s(t) &= \left(\sum_{k=0}^{\infty} (u_k \cdot \delta(t-kT)) \right) \bullet rect\left(\frac{t-T/2}{T}\right) \end{aligned} \tag{4-4}$$

$$rect\left(\frac{t}{T}\right) = \begin{cases} 1 & -T/2 < t < T/2 \\ 0 & OW \end{cases} \tag{4-5}$$

$$\begin{cases} x : \Re \rightarrow \Re^n \\ y : \Re^n \rightarrow \Re \end{cases} \quad \begin{pmatrix} A \in \Re^{n \times n} \\ B \in \Re^{n \times 1} \\ C \in \Re^{1 \times n} \\ D \in \Re \end{pmatrix} \tag{4-6}$$
$$\begin{cases} \dot{x}(t) = A \cdot x(t) + B \cdot u_s(t) \\ y(t) = C \cdot x(t) + D \cdot u_s(t) \end{cases}$$

Before going any further it is important to give some physical meaning to Eq. (4-6). The F&H presented in last section, relied on switching the current that charges (or discharges) a capacitor. The same method has to be applied to Eq. (4-6). In an active filter the state variables are invariably associated with a capacitor voltage. The capacitor voltage is defined as $dv(t)/dt = \dot{v}(t) = 1/C \cdot i(t)$, where C and $i(t)$ are the capacitor value and current, respectively. If the only dynamic element in the physical system defined by Eq. (4-6) is a capacitor, then accordingly $\dot{x}(t)$ represents a vector voltage derivative and the right hand side of Eq. (4-6) represents a current vector flowing through the capacitors. In order to make Eq. (4-6) a F&H representation, the integration just needs to be halted every T seconds for a time duration of $\tau - T$ seconds.

In a linear system, a generic representation for any instant of time referred to any other initial state instant at time t_0, can be stated as Eq. (4-7) where $e^{A \cdot T}$ is a state-transition matrix [61], evaluated for $t = T$.

If the system starts at rest (i.e., $x(t_0 = 0^-) = 0$), integrating for the first τ seconds results in Eq. (4-8). Remember Figure 4-8 where during T seconds:

$$\forall n \in \aleph, t \in (nT, (n+1)T]: u_s(t) = u_n.$$

$$\begin{cases} x(t) = e^{A \cdot (t - t_0)} \cdot x(t_0) + \int\limits_{t_0}^{t} (e^{A \cdot (t - \alpha)} \cdot B \cdot u_s(\alpha)) d\alpha \\ \\ y(t) = C \cdot x(t) + D \cdot u_s(t) \end{cases} \tag{4-7}$$

$$\begin{cases} x(\tau) = u_o \cdot \left(\int\limits_{0}^{\tau} (e^{A \cdot (\tau - \alpha)}) d\alpha \right) \cdot B \\ \\ y(\tau) = C \cdot x(\tau) + D \cdot u_0 \end{cases} \tag{4-8}$$

$$D0 = \{t \in \Re : \tau \le t \le T\}$$

$$(x(t) = x(\tau)), (y(t) = y(\tau)) \quad \forall t \in D0$$

This result is the initial condition for the next integration period that starts at $t = T^+$ and halts again at $t = T + \tau$. Remember that from Eq. (4-8) $x(t)|_{t_0 = T} = x(\tau)$, so reiterating Eq. (4-7) results in Eq. (4-9). Making the change of variable $\beta = \alpha - T$ on the second integral reduces Eq. (4-9) to Eq. (4-10). Iterating repetitively it can be easily proven (Appendix D) that the state and output of the system is represented by Eq. (4-11).

$$\begin{cases} x(T+\tau) = e^{A\cdot\tau}\cdot u_o\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\alpha)})d\alpha\right)\cdot B + u_1\cdot\left(\int_T^{(T+\tau)}(e^{A\cdot(T+\tau-\alpha)})d\alpha\right)\cdot B \\ \\ y(T+\tau) = C\cdot x(T+\tau)+D\cdot u_1 \end{cases} \qquad (4\text{-}9)$$

$$\begin{cases} x(T+\tau) = (e^{A\cdot\tau}\cdot u_0+u_1)\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\beta)})d\beta\right)\cdot B \\ \\ y(T+\tau) = C\cdot x(T+\tau)+D\cdot u_1 \end{cases} \qquad (4\text{-}10)$$

$$D1 = \{t\in\Re\text{:T+}\tau\le t\le 2T\}$$
$$(x(t)=x(T+\tau)),(y(t)=y(T+\tau)) \quad \forall t\in D1$$

$$\begin{cases} x((n-1)T+\tau) = \sum_{i=0}^{n-1} u_i\cdot(e^{A\cdot\tau})^{n-1-i}\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\beta)})d\beta\right)\cdot B & \forall n\in\aleph \\ \\ x(0^{+}) = 0 & n=0 \\ y((n-1)T+\tau) = C\cdot x((n-1)T+\tau)+D\cdot u_n \end{cases} \qquad (4\text{-}11)$$

$$Dn = \{t\in\Re:(n-1)T+\tau\le t\le nT,\ \forall n\in\aleph\}$$
$$(x(t)=x((n-1)T+\tau)),(y(t)=y((n-1)T+\tau)) \quad \forall t\in Dn, \forall n\in\aleph$$

The domain Dn results from the union of the sets defined as $D_i = \{t\in\Re:(i-1)T+\tau\le t\le iT\}$ for $i = 1, 2, \ldots n$, which are disjoint and consequently Eq. (4-11) does not define $x(t)$ for all time. Although it is easy to frame $x(t)$ for any instant of time it is genuinely not needed because the F&H output results from sampling $x(t)$ precisely at $t = nT$ which is in the domain Dn. Hence, sampling this equation every $t = nT$, results in the output discrete-time signal defined by Eq. (4-12).

$$y(n) = C\cdot\sum_{i=0}^{n-1} u_i\cdot(e^{A\cdot\tau})^{n-1-i}\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\beta)})d\beta\right)\cdot B+D\cdot u_n \qquad (4\text{-}12)$$

Now consider a second system represented by Eq. (4-13). The system is very similar to Eq. (4-6). Actually the only difference is that the feedback matrix A and the input vector B are scaled by k (for example by simply making the capacitor value to change by 1/k, ($k < 1$). In physical terms the current into the capacitor is k times smaller than in Eq. (4-6) Functionally this system is allowed to integrate during the full sampling and hold period T.

$$\begin{cases} z:\Re \rightarrow \Re^{n} \\ o:\Re^{n} \rightarrow \Re \end{cases} \qquad \begin{pmatrix} A \in \Re^{n \times n} \\ B \in \Re^{n \times 1} \\ C \in \Re^{1 \times n} \\ D, k \in \Re \end{pmatrix}$$
$$\begin{cases} \dot{z}(t) = k \cdot A \cdot z(t) + k \cdot B \cdot u_s(t) \\ o(t) = C \cdot z(t) + D \cdot u_s(t) \end{cases} \tag{4-13}$$

Applying the very same procedure utilized to find Eq. (4-11) at any instant of time $t = nT$, the response of this new system will be given by Eq. (4-14). Making the change of variable $\beta = (\tau/T) \cdot \alpha$ results into Eq. (4-15).

$$\begin{cases} z(nT) = k \cdot \sum_{i=0}^{n-1} u_i \cdot (e^{k \cdot A \cdot T})^{n-1-i} \cdot \left(\int_0^T (e^{k \cdot A \cdot (T-\alpha)}) d\alpha \right) \cdot B & \forall n \in \aleph \\ 0 & n = 0 \\ o(nT) = C \cdot z(nT) + D \cdot u_n \end{cases} \tag{4-14}$$

$$\begin{cases} z(nT) = \frac{T}{\tau} \cdot k \cdot \sum_{i=0}^{n-1} u_i \cdot (e^{k \cdot A \cdot T})^{n-1-i} \cdot \left(\int_0^\tau \left(e^{k \cdot A \cdot \frac{T}{\tau} \cdot (\tau - \beta)} \right) d\beta \right) \cdot B & \forall n \in \aleph \\ 0 & n = 0 \\ o(nT) = C \cdot z(nT) + D \cdot u_n \end{cases} \tag{4-15}$$

If now k is forced to be $k = \tau/T$ then Eq. (4-15) is finally reduced to Eq. (4-16), and the discrete-time output signal can in turn be defined by Eq. (4-17).

Comparing equations (4-16) and (4-17) with equations (4-11) and (4-12) respectively, it is possible to conclude that when $k = \tau/T$, Eq. (4-18) is verified.

Equation (4-18) states that the outputs of both systems are indistinguishable if sampled every $t = nT$. Thus, F&H filters are general. Given $k < 1$, it should be noted however, that the F&H defined by Eq. (4-11) would be utilizing k times smaller capacitors.

$$\begin{cases} z(nT) = \sum_{i=0}^{n-1} u_i \cdot (e^{A \cdot \tau})^{n-1-i} \cdot \left(\int_0^{\tau} (e^{A \cdot (\tau-\beta)}) d\beta \right) \cdot B & \forall n \in \aleph \\ 0 & n = 0 \\ o(nT) = C \cdot z(nT) + D \cdot u_n \end{cases} \tag{4-16}$$

$$o(n) = C \cdot \sum_{i=0}^{n-1} u_i \cdot (e^{A \cdot \tau})^{n-1+i} \cdot \left(\int_0^{\tau} (e^{A \cdot (\tau-\beta)}) d\beta \right) \cdot B + D \cdot u_n \tag{4-17}$$

$$[z(nT) = x(nT)] \Rightarrow [o(n) = y(n)] \tag{4-18}$$

Finally the difference equation that characterizes the F&H and the correspondent system Z transform is represented by Eq. (4-19) and Eq. (4-20) respectively (see Appendix E) and represents a step invariant filter [62].

$$x(n) = e^{AkT} \cdot x[(n-1)] + u_{n-1} \cdot [e^{AkT} - I] \cdot A^{-1} \cdot B; k = \tau/T \tag{4-19}$$

$$Y(z)/U(z) = (C \cdot [I - z^{-1} \cdot e^{AkT}]^{-1} \cdot z^{-1} \cdot [e^{AkT} - I] \cdot A^{-1} \cdot B + D); k = \tau/T \tag{4-20}$$

Previous qualitative and mathematical results have shown that the discrete-time F&H is a general procedure that can be applied to filters of any order. The filter may be low-pass, high-pass, band-pass or band-reject. In formal terms, a general F&H for a staircase discrete-time input followed by an ideal smoothing filter at the sampled output, can be stated as in Eq. (4-21). It should be noted however, that the physical system has a feedback matrix A and input vector B. The scaling term k ($k = \tau/T$) arises from the current switching method

$$H(W) = \frac{Y(W)}{U(W)} = \begin{cases} C\cdot[jW\cdot I-k\cdot A]^{-1}\cdot k\cdot B+D & |W|\le\frac{\pi}{T} \\ 0 & \text{O.W.} \end{cases} \qquad (4\text{-}21)$$

One might ask if any active filter implementation can be configured into a F&H system. There is no direct answer to this question. Some implementations will easily yield the F&H configuration, in others not so easily. The general rule is to start with a known physical active filter implementation and place switches strategically throughout the filter such that the current through the capacitors are periodically interrupted. However, for Eq. (4-11) and Eq. (4-16) to be equal, the output $y(t)$ has to be defined at the sampling times nT. In this sense the capacitor voltage (the state) has to be available at the output structure at all times in order to guarantee the correct sampling. This indicates that some active filter configurations might not meet these requirements.

Another interesting feature with the F&H is that no aliasing is introduced in the resulting system as Eq. (4-21) shows. The aliasing would occur due to the input sampling. However the input signal is assumed band-limited hence no distortion results due to aliasing.

4.4 Some Active Filters Simulation

Consider the general vector filter representation in Figure 4-9, and the low-pass KRC and multiple-feedback band-pass filters examples in Figure 4-10 [63]. Following previous discussion, switches were placed strategically to control the current. The sample and holds are added to sample the input (assumed band-limited) to generate $u_s(t)$ (Figure 4-8) and to synchronize the output to the input sampling clock.

In order to achieve a stable input, the F&H system and the sample and hold block are switched at different clock phases. The same is valid for the output (i.e., the output is sampled at a time instant where the F&H output is stable). Table 4-1 summarizes the parameter set used to simulate the vector filter. In addition, Table 4-2 summarizes the parameter set for the KRC and multi-feedback filters. The continuous-time prototype uses $1/k = 100$ bigger capacitors than the F&H.

Figure 4-11 to Figure 4-13 plot the simulation results of the different filter configurations depicted in Figure 4-9 and Figure 4-10. The sampling frequency is set to 20kHz in all F&H. Each plot shows the result for the F&H and the sampled and nonsampled outputs of the continuous-time filter. The input signal utilized for test is either a 250Hz square wave or the sinusoidal sum represented by Eq. (4-22).

$$\begin{aligned} u(t) &= \sin(2\pi200t + 0.056\pi) + 1.5\sin(2\pi330t + 0.185\pi) + \\ &\quad + 0.8\sin(2\pi2050t + 0.234\pi) + 1.3\sin(2\pi425t + 0.119\pi) \end{aligned} \tag{4-22}$$

As expected, both F&H and sampled continuous-time system outputs coincide. It can also be verified, in zoomed areas, that the non-sampled continuous-time output passes through the points defined at $t = nT$ with the same magnitude as the F&H. This is conformal with earlier theoretical developments.

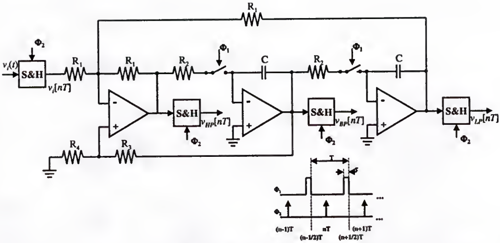


Figure 4-9. General vector filter

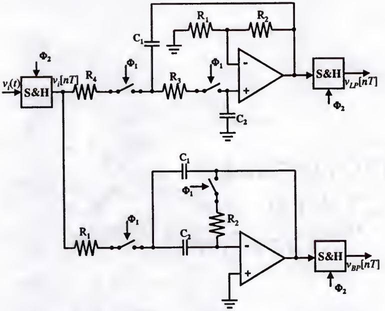


Figure 4-10. KRC low-pass (v_{LP}) and multiple-feedback band-pass (v_{BP}) filters

Table 4-1. Vector filter parameters

	Continuous-Time Vector Filter	**F&H Vector Filter**
R_1	10kΩ	10kΩ
R_2	10kΩ	10kΩ
R_3	86kΩ	86kΩ
R_4	10kΩ	10kΩ
C	15nF	**0.15nF**
$k = \tau/T$	-	1%
$f_0 = 1/(2\pi R_2 C)$	1061Hz	$\Leftrightarrow 1061Hz$
$Q = \frac{1}{3}\left(1+\frac{R_3}{R_4}\right)$	3.2	3.2

Table 4-2. KRC and multiple-feedback band-pass (BP) filter parameters

	Continuous-Time KRC Filter (Low-Pass)	**F&H KRC Filter (Low-pass)**	**Continuous-Time BP filter (High-Pass)**	**F&H BP Filter (High-pass)**
R_1	10kΩ	10kΩ	10kΩ	10kΩ
R_2	10kΩ	10kΩ	100kΩ	100kΩ
R_3	10kΩ	10kΩ	-	-
R_4	250kΩ	250kΩ	-	-
C_1	3.8nF	**0.038nF**	6nF	**0.06nF**
C_2	3.8nF	**0.038nF**	6nF	**0.06nF**
$k = \tau/T$	-	1%	-	1%
f_0	$1/(2\pi QR_3C_1)$ $= 834Hz$	$\Leftrightarrow 834Hz$	$1/(2\pi\sqrt{R_1R_2}C_1)$ $= 839Hz$	$\Leftrightarrow 839H$
Q	$\sqrt{R_4/R_3} = 5$	5	$(\sqrt{R_2/R_1})/2 = 1.6$	1.6

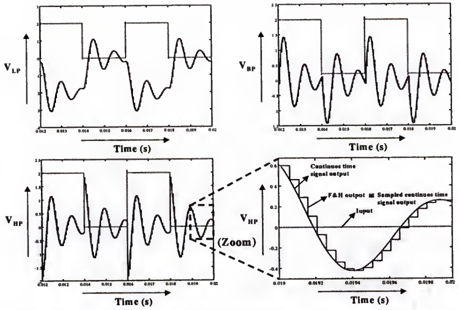


Figure 4-11. Vector filter response to a square input (continuous-time and F&H, see text for details)

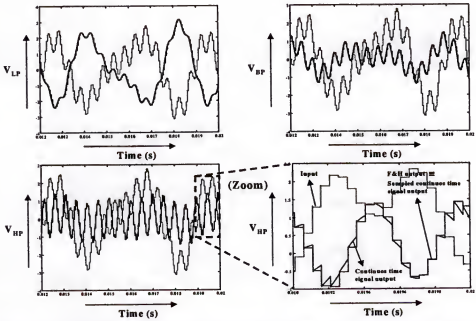


Figure 4-12. Vector filter response to a sum of sinusoids (Continuous-time + F&H, see text for details)

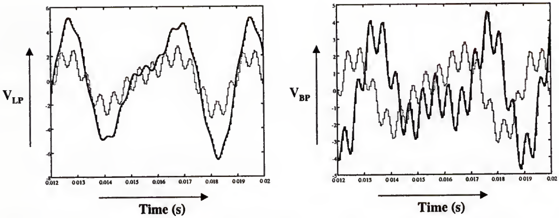


Figure 4-13. Simulation result for KRC and multiple-feedback band-pass filter of Figure 4-10 (the curves labeling is the same of Figure 4-12, and see text for details)

To test the generic discrete-time F&H representation defined by equations (4-19) and (4-20), a simulation was performed with Matlab using the vector filter in Figure 4-9. The state space representation of the vector filter continuous-time prototype is defined by Eq. (4-23). Making the feedback matrix A referenced as $A = \begin{bmatrix} -a & b \\ -c & 0 \end{bmatrix}$, the transition matrix will be given by Eq. (4-24) (with $\Upsilon = kT$ for Eq. (4-20)).

Figure 4-14 shows clearly that the F&H Hspice and the DSP Matlab simulation coincide, attesting the full dynamic characterization of the discrete-time F&H.

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \overbrace{\begin{bmatrix} -3 \cdot \dfrac{R_4}{R_3 + R_4} & \dfrac{1}{R_2 \cdot C} \\ -\dfrac{1}{R_2 \cdot C} & 0 \end{bmatrix}}^{A} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \overbrace{\begin{bmatrix} \dfrac{1}{R_2 \cdot C} \\ 0 \end{bmatrix}}^{B} u(t) \\ y(t) = \begin{bmatrix} y_{LP}(t) \\ y_{BP}(t) \\ y_{HP}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 3 \cdot \dfrac{R_4}{R_3 + R_4} & -1 \end{bmatrix} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} u(t) \end{cases} \qquad (4\text{-}23)$$

$$\left\{\begin{aligned}
e^{A\Upsilon} &= \begin{bmatrix} f_{11}\, f_{12} \\ f_{21}\, f_{11} \end{bmatrix} \\
f11 &= \left(\cos\left(\sqrt{bc-\frac{a^2}{4}}\cdot\Upsilon\right)-\frac{a}{2\cdot\sqrt{bc-\frac{a^2}{4}}}\sin\left(\sqrt{bc-\frac{a^2}{4}}\cdot\Upsilon\right)\right)\cdot e^{-a\Upsilon/2} \\
f12 &= \frac{b}{\sqrt{bc-\frac{a^2}{4}}}\cdot e^{-a\Upsilon/2}\cdot\sin\left(\sqrt{bc-\frac{a^2}{4}}\cdot\Upsilon\right) \\
f_{21} &= -\frac{c}{b}\cdot f_{12} \\
f_{22} &= f_{11}+\frac{a}{b}\cdot f_{12}
\end{aligned}\right. \qquad (4\text{-}24)$$

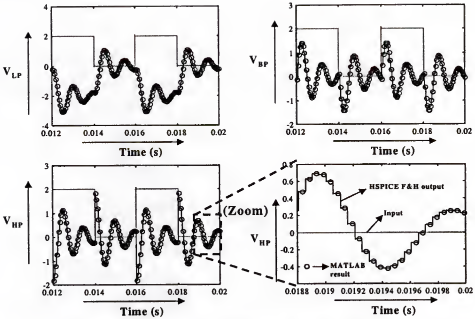


Figure 4-14. Discrete F&H vector filter response to a square input, versus MATLAB simulation of Eq. (4-19) for the same input

4.5 Continuous-Time F&H

The F&H with sample and hold inputs is, in signal processing terms, an implementation of a step invariant filter [62]. However, experiments have shown that the technique is more complex and powerful. Let *u(t)* be a generic continuous-time signal and band limited according to the Nyquist sampling criterion. This signal is then applied to the F&H input without any previous sampling. Experiments show that the ideally smoothed F&H output coincides with the signal coming from the equivalent continuous-time filter but, with scaled dynamics. If this holds for all band limited signals, then the S/H can be avoided. Nevertheless, a proof of generality is needed but, it is yet to be achieved.

Equation (4-25) represents a generic continuous-time system. The F&H counterpart is represented in Eq. (4-26) where *s(t)* is the switching signal defined by equations (4-4) and (4-5).

Figure 4-15 shows an Hspice simulation result for the vector filter defined by Eq. (4-23) (with parameter values listed in Table 4-1). The F&H output follows closely the continuous-time output signal. A Matlab Runge-Kutta simulation of Eq. (4-26) is also plotted in Figure 4-15. As expected, both Hspice and Matlab simulation results coincide, reinforcing the argument that Eq. (4-26) defines the physical F&H system.

$$\begin{cases}\dot{x}(t) = A\cdot x(t)+B\cdot u(t)\\ y(t) = C\cdot x(t)+D\cdot u(t)\end{cases} \tag{4-25}$$

$$\begin{cases}\dot{x}(t) = A\cdot x(t)\cdot s(t)+B\cdot u(t)\cdot s(t)\\ y(t) = C\cdot x(t)+D\cdot u(t)\end{cases} \tag{4-26}$$

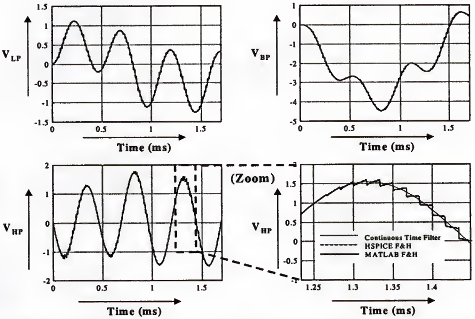


Figure 4-15. F&H vector filter response to a continuous-time signal found using Hspice and Matlab simulation

4.6 Noise in the F&H Filter

Without the mathematical framework necessary to analyze generic F&H systems with analog input signals, it is very difficult to analyze the noise performance. However it is possible to get some insights from first order analysis. In fact the first order F&H with a continuous-time input (Figure 4-16a) is a sample and hold circuit but, not restricted to $RC \ll \tau$. In fact it is exactly the opposite, the F&H relies in disregarding such a constraint to act as a mixed discrete-time filter. Nonetheless, the noise analysis for sample and holds [64] can still be extended for the first order F&H sections.

Consider then the low-pass F&H filter, and instead of a signal $V_i(t)$ take a voltage source $n(t)$ representing the total equivalent input noise. Noise in the present context is an uncorrelated stochastic signal with unlimited bandwidth and flat magnitude spectrum (i.e., the noise is white) and is mainly generated by the resistor. The spectral density is well known and given by Eq. (4-27) [64].

$$n(t)^2 = 4KTR \quad (V/Hz) \tag{4-27}$$

Assume that the filter is operating for a long time, long enough that it can considered working since $t = -\infty$ with the time reference chosen to be some place very far from the transient. Consider the low-pass and high-pass filter configurations shown in Figure 4-16. Let $X(jw)$ be the Fourier transform operation over $x(t)$. Finally, $F(s)$ represents an ideal smoothing filter with formal representation given by Eq. (4-28).

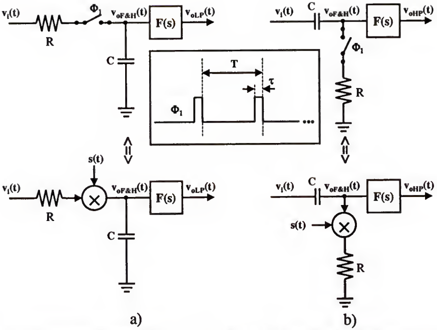


Figure 4-16. Continuous-time F&H (*F(s)* is an ideal smoothing filter and *s*(*t*) is a current signal). a) Low-pass; b) High-pass

$$F(jw) = \begin{cases} 1 & |w| \le \pi/T \\ 0 & |w| > (\pi/T) \end{cases} \tag{4-28}$$

The current in the capacitor can be defined as $s(t) \cdot i(t)$, where *i*(*t*) represents the current that flows when the switch is "on" and defined by: $i(t) = (v_i(t) - v_{oF\&H}(t))/R$. Then, the voltage at the output ($v_{oF\&H}(t)$) is given by Eq. (4-29), and the Fourier representation defined by Eq. (4-30). The *S(jw)* signal is a periodic pulse train with period $w_o = 2\pi/T$ with a *sinc*(.) envelope and defined by Eq. (4-31).

$$v_{oF\&H}(t) = 1/(RC) \cdot \int_{-\infty}^{t} \{s(t) \cdot (v_i(t) - v_{oF\&H}(t))\} dt \tag{4-29}$$

$$V_{oF\&H}(jw) = \frac{a}{jw} \cdot \{S(jw) \bullet (V_i(jw) - V_{oF\&H}(jw))\};\ \text{a}=\frac{1}{RC} \tag{4-30}$$

$$S(jw) = \tau \cdot \frac{\sin\left(w \cdot \frac{\tau}{2}\right)}{\left(w \cdot \frac{\tau}{2}\right)} \cdot \frac{1}{T} \cdot \sum_{m=-\infty}^{\infty} \delta(jw - mw_o) \Leftrightarrow$$

$$S(jw) = \frac{\tau}{T} \cdot \sum_{m=-\infty}^{\infty} \left(\frac{\sin\left(mw_o \cdot \frac{\tau}{2}\right)}{\left(mw_o \cdot \frac{\tau}{2}\right)} \cdot \delta(jw - mw_o) \right) \tag{4-31}$$

Finally, Eq. (4-30) can be rewritten for the noise source as shown in Eq. (4-32). *N(jw)* is the Fourier transform of *n(t)* and represents the total input noise. It further can be rearranged in form of Eq. (4-33). Since the noise is not band limited, the replicas in the summation term will foldover the base-band $|w| < w_0/2$. As expected, the left hand side of Eq. (4-33) also shows that the F&H output is replicated every mw_0 but modulated by the F&H low-pass transfer function for $m \neq 0$. However, if $w_0 >> a \cdot k$ the summation term will be strongly attenuated by the filter envelope and can be neglected. Thus, after an ideal smoothing filter is applied to the output (multiplying *F(jw)* on both sides of Eq. (4-33)), the base-band of the resulting signal is approximately a filtered aliased noise represented by Eq. (4-34).

$$V_{oF\&H}(jw) = \begin{array}{l} -\frac{ak}{jw}\left(\sum_{m=-\infty}^{\infty} \frac{\sin\left(mw_o \cdot \frac{\tau}{2}\right)}{\left(mw_o \cdot \frac{\tau}{2}\right)} \cdot V_{oF\&H}(jw - jmw_0) \right) + \\ \frac{ak}{jw}\left(\sum_{m=-\infty}^{\infty} \frac{\sin\left(mw_o \cdot \frac{\tau}{2}\right)}{\left(mw_o \cdot \frac{\tau}{2}\right)} \cdot N(jw - jmw_0) \right) \end{array} \tag{4-32}$$

$$V_{oF\&H}(jw) + \frac{ak}{ak+jw}\left(\sum_{\substack{m=-\infty \\ m \neq 0}}^{\infty} \frac{\sin\left(mw_o \cdot \frac{\tau}{2}\right)}{\left(mw_o \cdot \frac{\tau}{2}\right)} \cdot V_{oF\&H}(jw - jmw_0) \right) =$$

$$\frac{ak}{ak+jw}\left(\sum_{m=-\infty}^{\infty} \frac{\sin\left(mw_o \cdot \frac{\tau}{2}\right)}{\left(mw_o \cdot \frac{\tau}{2}\right)} \cdot N(jw - jmw_0) \right) \tag{4-33}$$

$$V_{oLP}(jw) \cong \frac{ak}{ak+jw} \cdot \hat{N}(jw) \tag{4-34}$$

Assume now an equivalent noise bandwidth, BW, constrained to $w_0 \ll BW < 2/\tau$ (the noise after the *sinc*(.) first zero is attenuated). To simplify further the analysis consider that $(2/\tau) \gg w_0$ (which is true for small duty-cycles) such that for $w \leq (2BW + w_0/2)$, $\operatorname{sinc}\left(mw_0\frac{\tau}{2}\right) \cong 1$ as Figure 4-17 shows. These conditions will maximize the aliased noise over the base-band frequency $|w| < w_0/2$. Therefore, the white noise that folds over the base-band will increase in magnitude as much as the equivalent bandwidth decrease (Figure 4-18). This is the signal $\hat{N}(jw)$ in Eq. (4-34) that results after the ideal smoothing filter. Consequently the noise increases in base-band due to undersampling but, is also reshaped by the equivalent F&H filter. Similar result is obtained with the first order high-pass F&H.

Similar effects occur with sampled systems other than F&H. However, since F&H bent more for low frequency poles applications, the equivalent resistors will possibly be larger. Therefore, in general a F&H system will render a higher base-band noise spectral density than systems that do not rely on large resistor values to accomplish the time constant.

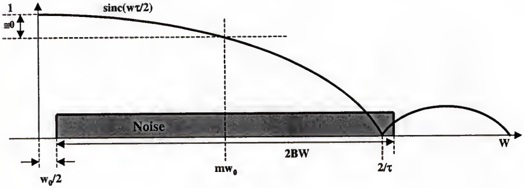
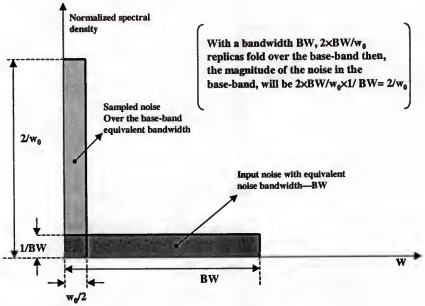


Figure 4-17. Modulated noise replicas


Normalized spectral density
Sampled noise Over the base-band equivalent bandwidth
$2/w_0$
Input noise with equivalent noise bandwidth—BW
1/BW
BW
w
$w_0/2$

With a bandwidth BW, $2\times BW/w_0$ replicas fold over the base-band then, the magnitude of the noise in the base-band, will be $2\times BW/w_0\times 1/BW = 2/w_0$

Figure 4-18. Input and sampled noise spectral densities in the frequency domain

The F&H is a sampled data system and consequently it is also sensitive to clock feedthrough. Harmonic distortion and offsets resulting from CFT will influence the filter performance. For low-frequency applications however, the capacitors used in the F&H system should be reasonably high thus reducing the charge error.

4.7 F&H VLSI Implementation of K0 Dynamics

The F&H circuit designed for the K0 dynamics is shown in Figure 4-19. The circuit is a cascade of two first order low-pass filters. Each low-pass section is updated in different phase clocks allowing to internally integrate a delay block necessary to uncouple instant feedback when this block is applied to built the silicon olfactory model.

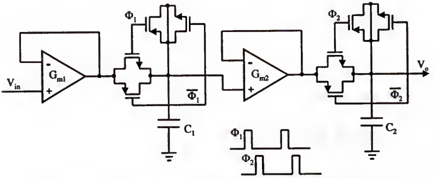


Figure 4-19. F&H implementation for the K0 dynamics

The G$_m$ block in Figure 4-19, is an active implementation of a resistor [65]. It is a simple differential amplifier stage with emitter degenerator MOSFET diodes as shown in Figure 4-20. These diodes decrease the G_m value for the same bias current and also doubles the input linear dynamic range. The relation between the input differential voltage and de output current is defined by Eq. (4-35). This is a nonlinear relationship however if the input difference, $(V_{in+} - V_{in-})$ (Figure 4-20), is kept small, it is reasonable to take the derivative of the *tanh*(.) around zero resulting in Eq. (4-36).

$$I_{out} = I_{bGm} \cdot \tanh\left(\frac{(V_{in+} - V_{in-})}{4V_T}\right) \tag{4-35}$$

$$I_{out} = \frac{I_{bGm} \cdot (V_{in+} - V_{in-})}{4V_T} \tag{4-36}$$

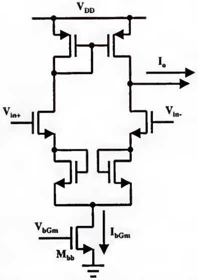


Figure 4-20. G_m amplifier with degenerator diodes

Equation (4-37) represents the Z domain transfer function for Figure 4-19. The very low frequency poles (35Hz and 114Hz) were accomplished with a very low-power implementation ($\cong 100nW$) and with a fairly low area expenditure of $200\mu m \times 150\mu m$ ($C_1 \cong 1pF; C_2 \cong 4pF$). The duty-cycle was set to 1% virtually scaling up the capacitors by 100. In Figure 4-21 the predicted and measured frequency responses are shown.

$$H(z) = \frac{(1 - e^{-(G_{m1}/C_1)kT})}{1 - e^{-(G_{m1}/C_1)kT} \cdot z^{-1}} \times \frac{(1 - e^{-(G_{m2}/C_2)kT})}{1 - e^{-(G_{m2}/C_2)kT} \cdot z^{-1}} \times z^{-1} \tag{4-37}$$

The poles location, defined by Eq. (4-37), is linearly dependent on the sampling period T and duty-cycle k. Measured results shown in Figure 4-22 display that linear relationship. However, the slope is not exactly the same on both plots. The difference is about 33% and is mostly due to duty-cycle errors.

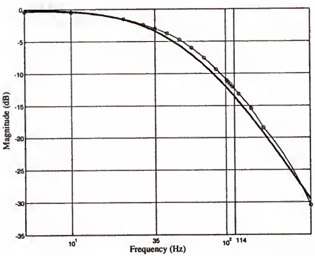
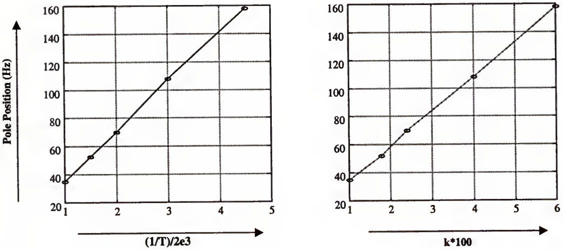


Figure 4-21. F&H measured frequency magnitude response together with the response predicted by (4-37) (solid)



a) b)

Figure 4-22. Dominant pole position. a) Pole variation versus normalized sampling frequency; b) Pole location versus normalized duty-cycle (k).

4.8 Summary

In this chapter, a new class of filters suited for the low frequency dynamics of the olfactory model was proposed. The filter is denominated “Filter and Hold (F&H)” and it was shown that the time constant $\tau = RC$ is virtually increase by $1/k$ through time manipulation. The VLSI design and measurements for the K0 dynamics were also presented.

CHAPTER 5
RESULTS

5.1 Introduction

The high level formal simulation proposed in chapter 2 became the center piece for the discrete-time analog implementation. At this point the formal high level representation can, in a sample by sample basis, predict how the real system will behave. Thus, the VLSI system can be flexibly tested and the parameters refined with the digital simulator which can be directly brought onto the designed chip by external active control. This is possible because both domains share common mathematical descriptions (difference equations and Z transform), except that the DSP representation is dimensionless. However, this does not pose a problem because the variables are easily translated (e.g., index time n in DSP becomes *nT* in the real domain, and amplitudes are scaled). Of course this tight coupling between the mixed signal and the DSP implementations also creates extra difficulties because the exact mathematical functions are seldom implementable in analog VLSI and the present implementation is not an exception. Hence, the design cycle has to include a redefinition of the formal DSP system to include the alterations imposed by analog implementation limitations. If it turns out that both representations have a close fit, the analog implementation will be a good representation of the original continuous-time model, given that previous results have shown that the digital model reproduces well the Runge-Kutta simulation. This is the methodology followed in this chapter that culminates previous results in a consistent framework.

The last chapter presented a dynamical system that made possible the implementation of the K0 dynamics with very low power and reasonable area. However, two more blocks need to be designed: the nonlinear function and the summing node, to complete the K0 cell. This is the subject of the next two sections.

5.2 Nonlinear Asymmetrical Sigmoid Function

The function responsible for the nonlinear characteristic of Freeman's olfactory model, is static and asymmetric around the zero point (Figure 1-1). A precise implementation of Eq. (1-2) is not an easy task but, can be approximated with a modified OTA (operational transconductance amplifier). The exponential sigmoid shape and the asymmetry around the input axis present in the original function, are preserved with this design. The exponential shape is set by placing the MOS transistor in subthreshold operating region where it displays an exponential relationship between the gate source voltage and the drain current. An asymmetric characteristic is accomplished by unmatching the current mirror that acts as an active charge to a differential input stage and shown in Figure 5-1. The output current relation with the input differential voltage is then represented by Eq. (5-1), and is plotted in Figure 5-2 for $I_b = 1$; $N = 5$; $V_T = 1$.

$$Q(V_{in}) = I_b \times \frac{N \cdot e^{(V_{in}/VT)} - 1}{e^{(V_{in}/VT)} + 1} \qquad (5\text{-}1)$$

The unbalancing of the current mirror originates the offset displayed in Figure 5-2a. However, it can be cancelled by attaching a voltage equal to the offset value at the cell inverting input. This voltage value can be automatically achieved with a similar cell connected with negative feedback. The compensated nonlinear function block diagram is shown in Figure 5-3. A final residual offset will be only caused to the natural fabrication process.

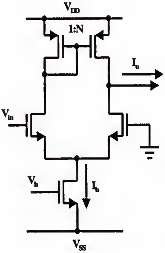
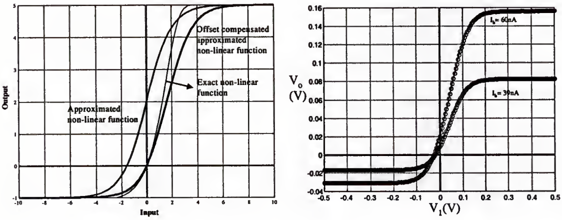


Figure 5-1. Approximated nonlinear function implementation



a)

b)

Figure 5-2. Approximated nonlinear function. a) N=5, I_b=1, VT=1; b) Measured responses in the chip for two different bias currents (the bias current of G_m is I_{bGm}=600nA)

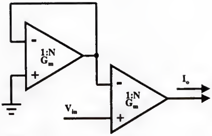


Figure 5-3. Asymmetric nonlinear function with offset compensation

The final nonlinear function schematic is shown in Figure 5-4. The current mirror gain N, was set to 5 which is equivalent to make q_i=5 in equation (1-2). The OTA was implemented using a wide range topology, to prevent the current inversion that occur with single stage differential amplifiers, when the output voltage drops below a certain value. The G_m amplifier is a current to voltage converter. Chip measurements [66] with two different bias currents are shown in Figure 5-2b.

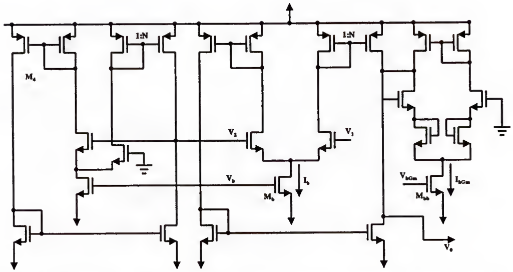


Figure 5-4. Implemented asymmetric nonlinear function (voltage in/voltage out)

5.3 The K0 Cell

The K0 model is almost complete. To allow interconnectivity, an input summing node is required and also excitatory and inhibitory K0 cells are needed. The summing node is implemented with a simple inverting voltage adder as represented in Figure 5-5a. In order to preserve the proper sign between input and output, the nonlinear circuits were topologically changed to preserve the signal polarity. Figure 5-5b shows conceptually the needed nonlinear function change to ensure the proper signs and to accomplish the K0 excitatory and inhibitory cells. The circuit to perform these nonlinear functions are a small variant of Figure 5-4 circuit. Only the unbalancing transistors and input positions change. Each designed K0 takes an area of $500\mu m \times 160\mu m$ and the power consumption is roughly $200\mu W$ to ensure driving capabilities in the summing and output amplifiers.

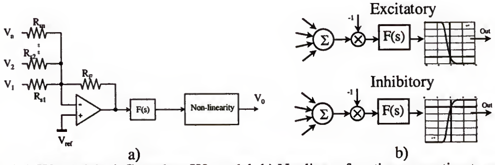


Figure 5-5. K0 model. a) Complete K0 model; b) Nonlinear function correction to accommodate the signal inversion at the summing node

5.4 Preliminary Results

One main objective of this dissertation is to present implementation methods that can efficiently be used for physical analog implementations. To prove this, implementation examples are needed. All crucial elements were designed and tested, however the controlling digital signals were generated externally considering that this part is not the focus of this thesis. Furthermore in a full integration these would be library ASIC components. The other external elements were the resistive feedback strengths. This is an important component and should be integrated, however to facilitate the testing (the parameters were not known a priori) these were externally placed. Possible solutions will be the object of discussion in the next chapter, the future work section.

5.4.1 A Single KII Oscillator

A KII set was built in analog VLSI with components described in the last sections and with the F&H described in last chapter. The parameter set was empirically determined. In opposition with what would be the normal procedure, the DSP simulation confirmation (NeuroSolutions or Matlab) is done after the chip test using the empirical parameters found experimentally. In fact, in terms of validation, it does not matter which method is done first, consequently, it is not relevant which system is used as a parameter reference. Starting with a working circuit provides a measure of how close the digital simulator proposed in chapter 2 predicts the response and in the case of an unsatisfactory solution, it will assist in the search of what components should be modified. All in all, this testing procedure helps to fine tune the digital simulator to the analog implementation.

The circuit and simulator configurations are shown in Figure 5-6a and Figure 5-6b, respectively. Table 5-1 summarizes the parameter set used in this experiment. The K0 linear dynamics in these experiments had poles set at $a = 110/s$ and $b = 360/s$, and the nonlinearity was biased with $I_b = 40nA$ (Figure 5-2). This biasing value was set empirically and is such that the output is well below the input saturating levels of the F&H G_m amplifiers.

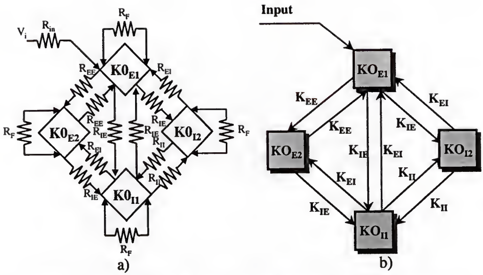


Figure 5-6. The KII model. a) Circuit schematic; b) DSP counterpart

Table 5-1. KII interconnection strengths

Chip and digital parameter values

Gain	Value
$R_F/R_{EE}=K_{EE}$	2.5
$R_F/R_{II}=K_{II}$	2.5
$R_F/R_{IE}=K_{IE}$	2
$R_F/R_{EI}=K_{EI}$	5

The clock frequency is set to $f_s = 2.3kHz(> 2 \cdot \pi/(20 \times b))$, the duty-cycle $k = 1\%$ and $R_F = 100k\Omega$. As done in earlier chapters, an output is referred to the state of the system. We normally define it at the dynamics output.

A KII cell is an oscillator controlled by the input. The KII cell oscillates when the input energy rises and ceases oscillation when the input goes below a certain value. Figure 5-7 plots the measured response at the KO_{E1} output evidencing such behavior. In Figure 5-8 signals measured at different KO cells in the KII are plotted.

On a first attempt, the digital simulator with feedback parameters defined by Table 5-1 does not show a satisfactory solution. After taking the system out of the initial zero state, Figure 5-9a shows that the output never ceases oscillation with the same input used in Figure 5-7. This result clearly shows that the original DSP blocks are not appropriate representations of the implemented VLSI approximations. It is then convenient to redefine some of these digital simulator blocks responsible for the differences to take into account the major necessary approximations made through the VLSI design.

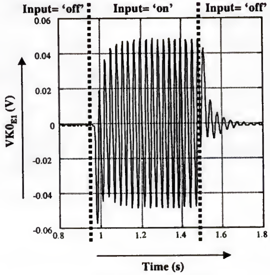


Figure 5-7. KII chip measurement response to a pulse input

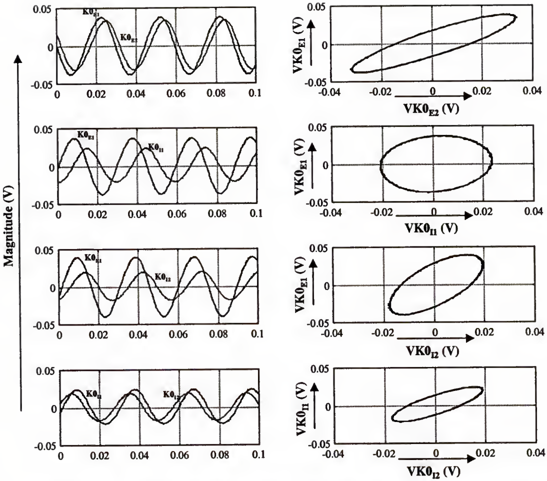
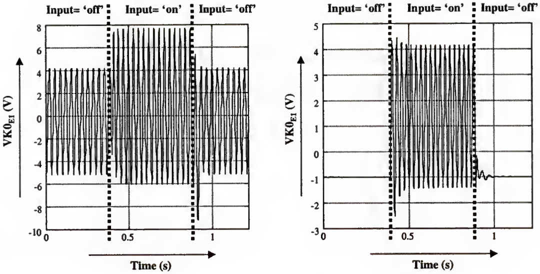


Figure 5-8. KII measurement at different outputs during the oscillating phase



a) b)

Figure 5-9. Simulator response with. a)- Exact nonlinearity; b) Exact nonlinearity with normalized input and corrected slope

A more careful analysis of Figure 5-2a reveals that the exact nonlinear function proposed by Freeman has a higher slope and consequently requires a smaller input for the same saturating output levels. Clearly this a factor that makes a difference. In fact, the normalization parameter α needed to match the slopes of the two functions is about $\alpha = 1.5$. Furthermore, the implemented nonlinearity has an input normalization factor of $V_T = 25mV$, and a set-up positive saturating level of 80mV. To scale up this function to the -1 and 5 saturating levels used in the digital simulations of Freeman's nonlinearity, we have to make in Eq. (5-1) $I_b = 1$. The resulting normalizing factor becomes $V'_T = 5/(80 \times 10^{-3}) \times 25 \times 10^{-3} = 1.56$. This scaled up function has the same saturating levels as the original Freeman's nonlinear function with $q = 5$, but with different slope due to the different normalization of $V'_T = 1.56$. If now we correct Freeman's nonlinearity slope with an input normalizing factor of $\alpha' = \alpha \times V'_T = 1.5 \times 1.56 = 2.34$

and perform a digital simulation in the same conditions as before, a more satisfactory result is obtained except for the obvious scaled up amplitudes as shown Figure 5-9b. The result is coherent and shows clearly that in the future, the K0 nonlinear function needs to be reprogrammed to take into account the approximated functions implemented in the analog VLSI chip.

Taking in consideration the approximated nonlinear function and also the F&H difference equation, a simulation result in the same condition as the chip measurement is shown in Figure 5-10 and Figure 5-11. Comparing these figures with Figure 5-7 and Figure 5-8 the similarities are evident. The main measurable features are summarized in Table 5-2. The relative phase displacement and magnitude ratios are coherent although, with different absolute values. It should be noted that the system is of eighth order, which means that the frequency difference (34Hz given by the simulator and the 38Hz measured from the chip) can account for as much as 20^{o} phase difference and 2 times in magnitude. In this sense the measurements are consistent.

The major sources of error responsible for the remaining differences observed in Table 5-2 is the F&H implementation and other nonidealities associated to all other components in the K0 chain. Starting with the G_m stage in Figure 4-19 when the switch is “on” the resulting differential equation is given by Eq. (5-2). This is clearly a nonlinear function. Although the difference $v_{in} - v_o$ should be kept small, the fact is that assuming G_m the derivative of Eq. (5-2) around $v_{in} = 0$ is an approximation.

$$C \cdot \frac{dv_o}{dt} = I_b \tanh\left(\frac{v_{in} - v_o}{4V_T}\right) \tag{5-2}$$

The nonlinear hyperbolic tangent function will have a re-shaping effect on the signals and will effectively limit the signals magnitudes. Other higher order effects such as clock feedthrough and extra delays due to finite bandwidth of all the components have not been accounted in the digital representation. Together with all the nonideal effects, is the usual parameter uncertainty that is always associated with real implementations and that affects all components. Albeit these natural and unavoidable errors (although perhaps minimizable), the discretization methodology shows to be a good prediction of how the mixed signal analog system behaves dynamically, and will be an asset for the analog circuit parametrization.

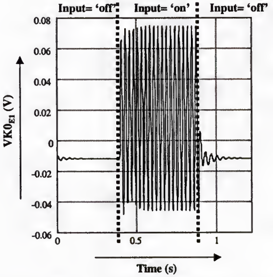


Figure 5-10. KII digital simulator result to a pulse input

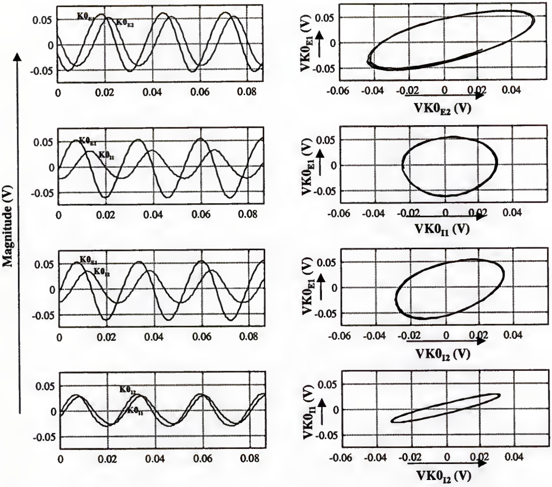


Figure 5-11. KII simulator result at different outputs during the oscillating phase

Table 5-2. Some important results from simulation and chip measurement (S- represents the state magnitude)

Feature	**Simulation**	**Chip Measurement**
S_{KE1}/S_{KE2}	1.15	1.15
S_{KE1}/S_{KI1}	1.71	1.69
S_{KE1}/S_{KI2}	2.18	2.00
S_{KI1}/S_{KI2}	0.89	1.28
$\Phi(K_{E1}, K_{E2})$	42.6^{o}	30.4^{o}
$\Phi(K_{E1}, K_{I1})$	86.6^{o}	85.1^{o}
$\Phi(K_{E1}, K_{I2})$	61.8^{o}	48.7^{o}
$\Phi(K_{I1}, K_{I2})$	-23.4^{o}	-36.5^{o}
Frequency	38Hz	34Hz

5.4.2 The Reduced KII

When the number of channels in a KII (or KIII) network increases in number, two of the K0 cells, more specifically KO_{E2} and KO_{I2} in Figure 5-6, become redundant [15]. Removing these two cells leads to the reduced KII shown in Figure 5-12. An experiment was performed on this cell. The parameter set used in both chip experiment and the digital simulator can be found in Table 5-3. In Table 5-4 some important dynamic measures are summarized and Figures 5-13 and 5-14 plot together the measured and the predicted responses by the high-level formal simulation.

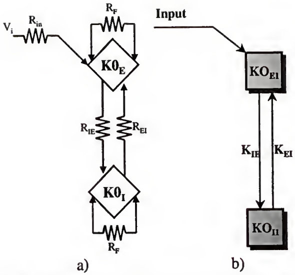


Figure 5-12. Reduced KII. a) Circuit schematic; b) DSP counterpart

Table 5-3. Reduced KII interconnections strengths

Chip and digital parameter values

Gain	Value
$R_F/R_{IE}=K_{IE}$	2
$R_F/R_{EI}=K_{EI}$	7.7

Table 5-4. Some important results from simulation and chip measurements (S- represents the state magnitude)

Feature	Simulation	Chip Measurement
S_E/S_I	2.33	2.14
$\Phi(K_E, K_I)$	95.8°	92.2°
Frequency	31Hz	32Hz

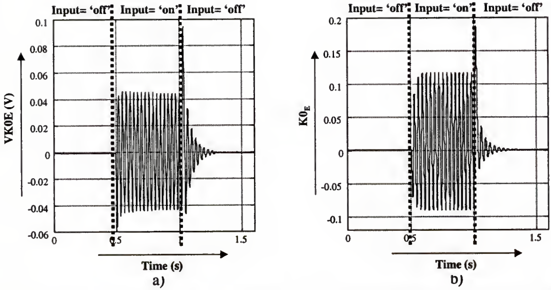


Figure 5-13. Reduced KII response to a pulse. a) Chip measured response; b) DSP simulation

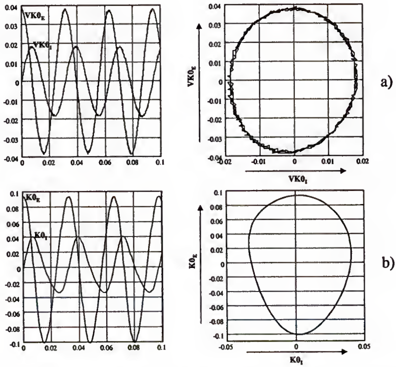
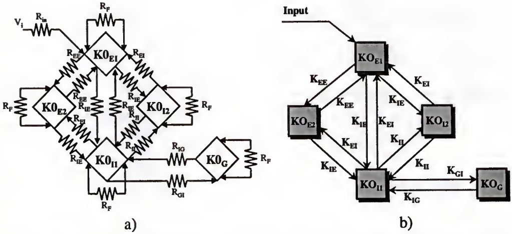


Figure 5-14. Steady-state oscillation response. a) Chip measurement; b) DSP simulation

Previous results show that the normalized dynamical characteristics are similar, however there are observable differences that can be explained with the same arguments used in the last section.

5.4.3 A Coupled KII–K0 Cell

The last results have shown that the simulator can reproduce well the dynamics of a real discrete-time analog VLSI implementation. A further step was taken and a K0 extra cell was coupled to a KII as shown in Figure 5-15. This configuration topologically represent the last two layers of KIII (Figure 1-8). The parameter set used in the experiment is summarized in Table 5-5. The results are shown in Figure 5-16 and Figure 5-17 and reinforce previous arguments.



a) b)

Figure 5-15. Coupled KII-K0. a) Circuit schematic; b) DSP counterpart

Table 5-5. Coupled KII-K0 interconnection strengths

Chip and digital parameter values

Gain	Value
$R_F/R_{EE}=K_{EE}$	2.5
$R_F/R_{II}=K_{II}$	2.5
$R_F/R_{IE}=K_{IE}$	2
$R_F/R_{EI}=K_{EI}$	5
$R_F/R_{GI}=K_{GI}$	3.3
$R_F/R_{IG}=K_{IG}$	2

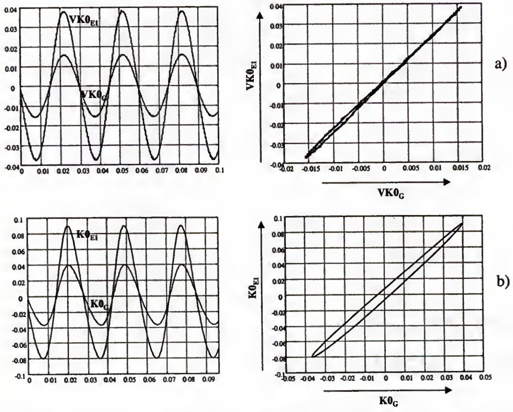


Figure 5-16. Steady-state oscillation response. a) Chip measurement; b) DSP simulation

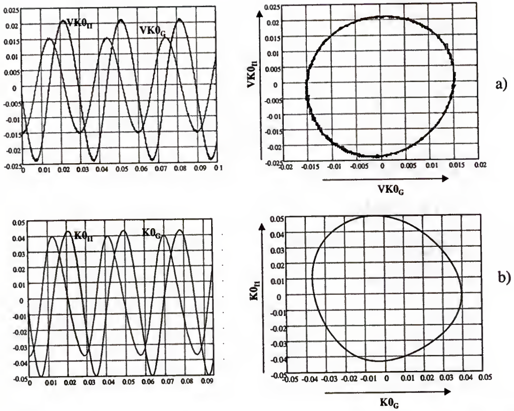


Figure 5-17. Steady-state oscillation response. a) Chip measurement; b) DSP simulation

5.5 The KII Network

A final experiment is needed to prove another important argument in this dissertation. Previously, we stated that the discrete-time implementation allows transparent multiplexing due to the system inherent time synchronization. Transparent means that the system operates seemingly as if it had access to all the resources at all times. It is clear that, for a continuous-time system, multiplexing effectively samples the signals. To keep the system coherence these sampled signals need to be filtered. However, if the system is already sample type, the dead time in-between sampling instants can be used to multiplex resources.

A two phase clock is almost always needed in analog sampled systems. One phase to sample the input and another to update the system output. The first phase can be multiplexed and the outputs updated at the same instant on the second sampling phase to keep the same time reference. With this method, a dramatic reduction of the interconnectivity size is possible for the OB and PG layers of the KIII olfactory model. This is the theme of the following sections

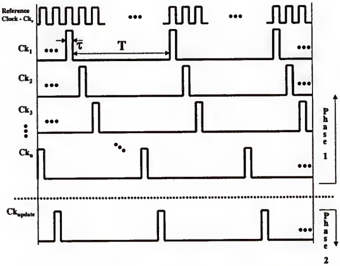


Figure 5-18. Multiplexing time clocks

5.5.1 Network Structure Memory and Multiplexing

For simplicity, the multiplexing concept is going to be presented using reduced KII cells at the OB layer. Extending the method to the regular KII and KIII is immediate. The first clock phase (Φ_1 in Figure 4-19) is divided into n out of phase similar clocks (with the same duty-cycle) as shown in Figure 5-18. Each clock signal Ck$_i$ ($i = 1, 2, \ldots, n$) is used to update the F&H first stage of channel i as seen in Figure 5-19. This stage works both ways, filters, and holds the signal, for a final update on the second stage originating the necessary delay. After the sequential input update is ended, all F&H output cells are updated simultaneously on the second phase (Ck_{update} on Figure 5-18 and Figure 5-19). The cell output is then stable and ready to be used on another input update cycle. Thus, the clock method represented in Figure 5-18, separates the update of each cell which is significant for the multiplexing method. Figure 5-19 shows an example where n KII channels are multiplexed. In this particular example the second channel would be the one to be updated (i.e., Ck_2 is active). Note that both excitatory and inhibitory K0 cells correspondent to channel 2, are active. The other channels are all open, thus there is no interference in-between channels. The feedback is common to all cells but, since only one channel is active at a time, the common feedback works as individual feedback on each input sequential update.

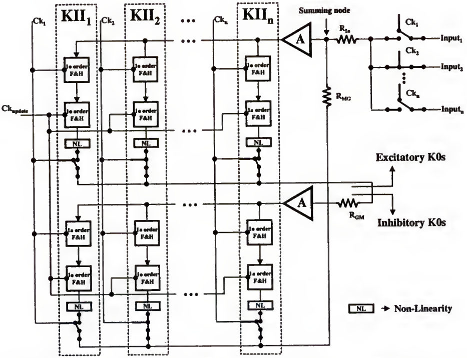
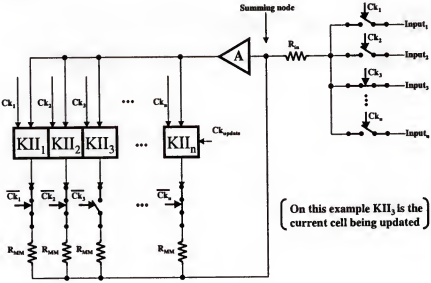


Figure 5-19. Multiplex example of n reduced KII cells (in this example Ck_2 would be active and KII_2 is the current updating cell)

The unique feedback is possible because individual cells share the same parameter set, a single gain component K_{MG} and K_{GM} is needed. These gains are represented by R_{MG}, and R_{GM} in Figure 5-19 respectively. The number of gain components is then reduced from $2 \cdot n$ to 2 in a reduced KII network, and from $10 \cdot n$ to 10 in a full KII network.

To build the KII network, the current cell being updated needs to have access to all other cells output. Since the update is not simultaneous the interconnections among cells can effectively be multiplexed and the number of interconnections can be reduced from $2 \cdot (n^2 - n)$ to $2 \cdot n$ [67]. Conceptually the procedure is shown in Figure 5-20 for the excitatory interconnections. Each KII_i cell represents the correspondent dashed block in Figure 5-19. A second set of switches connections are added and stay always "on" at all times except the switch corresponding to the current cell update (cell number 3 in Figure 5-20 example) because auto-feedback is not allowed. The inhibitory interconnectivity follow the same concept.



On this example KII_3 is the current cell being updated

Figure 5-20. Multiplex example of the excitatory interconnections among KII cells in a KII network

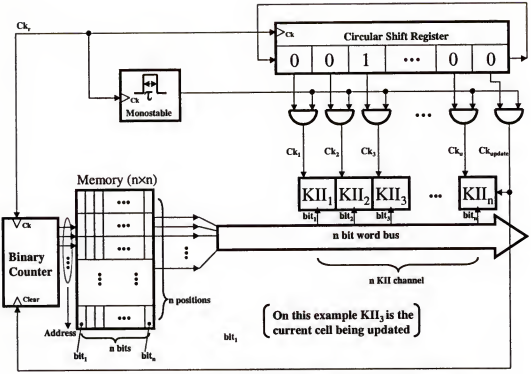


Figure 5-21. Clock generation and network memory setup

It should be emphasized that the sequential update has no effect on the dynamical behavior when compared to a simultaneous update of all cells. This is so because of the two phase clock update needed to accomplish the necessary delay. There is an input and output update phases that take τ seconds to update (clock signal with $k=\tau/T$ duty-cycle). Out of these time intervals the signals are constant. The multiplex just uses the stop time between sampling intervals to interleave the input update phase among the cells. Keeping the output update simultaneous shyncronizes the signal time references.

Finally for the network to function as an associative memory, the set of gain parameters K_{MM} (R_{MM}) have to change from cell to cell update, according to the Hebbian rule discussed in earlier chapter. Furthermore, at the update time of a particular cell, that cell needs to have the correct feedback parameters set according to the stored patterns. Thus, an external memory to the system is needed to store the network topology that each cell needs to see at the corresponding update time. This memory array is square (i.e., it needs to have n bits and n memory positions, as many as the number of network channels) and the content is binary because the possible number of different values is two (the interconnection is set to an high or low value gain). This solution is depicted in Figure 5-21.

Figure 5-21 also shows generically how the different clock signals are generated. The cells clock signal origin is common to all cells, (a monostable) that is triggered by a reference clock (Figure 5-18). The common clock signal minimizes the timing errors in-between cells. The cell is then chosen according to a select signal coming from a circular shift register where the "1" position defines the active cell for input update.

5.5.2 Four Channels KII Experiment

The experiment presented in this section serves to prove the concept described above, and also alert for some other improvements to future implementations. The simplified setup is shown in Figure 5-22. The KII network is composed of four channels and the multiplexing block represents the last section method. A four channel network has very little capacity concerning possible different patterns that can be stored. So, a single pattern was stored and set to [1 0 1 0]. Consequently the memory content in Figure 5-21 needs to be set as shown in Figure 5-23. The individual KII oscillators were set by the parameters defined in Table 5-1, however, the remaining parameters were empirically found with the help of the DSP simulator. The dynamic parameters were set to $a = 220/s$ and $b = 720/s$. After few tentatives with different parameters a satisfactory result was obtained with the set found.in Table 5-6.

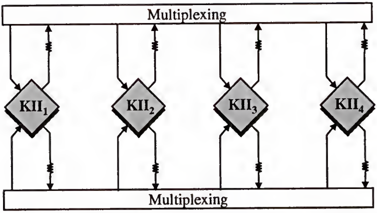


Figure 5-22. Simplified representation of the four channels KII experiment

Cell output gain (× - deactivated)

Address to current cell	KII_1	KII_2	KII_3	KII_4
KII_1	×	0	1	0
KII_2	0	×	0	0
KII_3	1	0	×	0
KII_4	0	0	0	×

Figure 5-23. Memory set up for the pattern [1 0 1 0]

Table 5-6. In-between KII cells interconnections strengths

Chip and digital parameter values

Gain	Value
$R_F/R_{EEL}=K_{EEL}$	0.5
$R_F/R_{EEH}=K_{EEH}$	2
$R_F/R_{II}=K_{II}$	0.25

An input is applied to the network corresponding to the stored pattern. The result is found in Figure 5-24. As expected, the channels that have the input applied evolve into a higher magnitude oscillation pattern while the others oscillate but with much smaller magnitude. Definitely it is possible to be recognized as a [1 0 1 0] pattern. It should be pointed out however, that it is not easy to replicate the exact digital simulations because the KII individual cells are never alike in an analog implementation. The consequence is different impulse response and natural frequencies between cells. This is noted on the response displayed by channel 1 and 3 in Figure 5-24. The channels respond differently to the transient though, locking to the same frequency. Eventually, if the cells are to far apart in terms of their natural oscillating frequencies, the lock between the KII channels may never happen. Figure 5-25 shows another effect not expected from ideal equal cells. Channels coupled with connections to form a pattern lock in phase however, a deviation is observed. This deviation means that the signals do not have the exact same shape.

The main block responsible for such differences is the nonlinear function that had to be setup individually on each cell and care should also be taken concerning equalization of the offsets. Although not implemented, future designs of KII cells should multiplex the nonlinear block between all K0s in order to make the dynamic behavior uniform, but more importantly to facilitate the network setup. If many channels were in use it would become impractical to adjust each cell individually. Nonetheless it is shown that multiplexing is correct, important and indispensable to implement realistic KII network sizes.

The DSP result with the same setup is shown in Figure 5-26 and Figure 5-27. It shows a similar response to the chip although, as expected, it does not show deviations in the phase plot because the cells are exactly equal. The DSP signal has a 61Hz frequency while the chip measurements present 63Hz (although it can be adjusted to 61Hz by changing one of the factors that affect the F&H poles position).

Finally, with the parameter set used, the input signal had to go below zero to cease the oscillation. If not, the KII output would never rest into a stable value with zero input. This is was verified in both DSP and implemented analog chips.

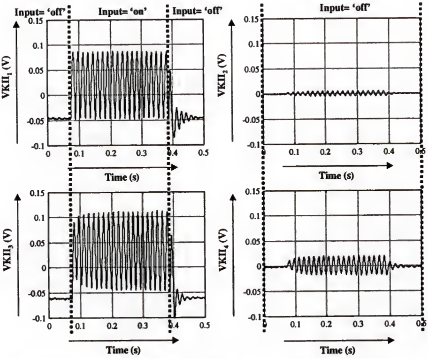


Figure 5-24. Measured 4 channel multiplexed KII network

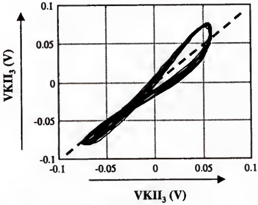


Figure 5-25. Channel 3 vs. channel 1 phase plot

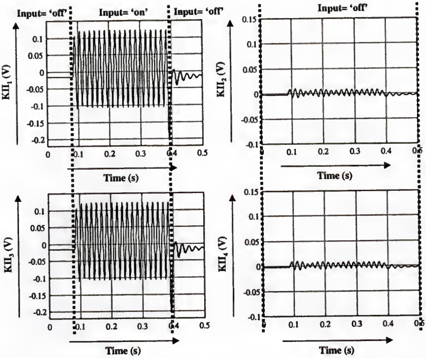


Figure 5-26. The DSP simulation measure of a 4 channel KII network

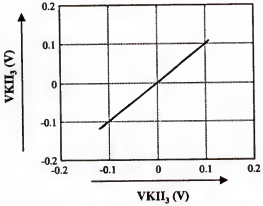


Figure 5-27. Channel 3 vs. channel 1 DSP simulation phase plot

5.6 Summary

This chapter presented results that culminate the DSP simulator and the analog implementation in a consistent framework. It was shown that the DSP and chip measurements can, in essence, predict each other‘s behaviors. The observable differences were also discussed and a multiplexing methodology was presented to reduce the number of needed resources.

CHAPTER 6
CONCLUSIONS AND FUTURE DIRECTIONS

6.1 Conclusion

This dissertation has presented a design and implementation framework of a biologic realistic model of the olfactory cortex proposed by Freeman. It first started with a digital mapping of the continuous model. Albeit an approximation, the digitizing method was shown to be effective and yields digital models that describe sufficient well the complex behavior of the olfactory system to allow parameter fine tuning. The method of discretizing the model is less computational intensive than Runge-Kutta numerical procedures, but its most important feature is flexibility. To study complex systems, one has to be able to interact with the simulator and eventually change parameters and topology. With the digital model these changes are easily accomplished considering that all elements in the network have been individually modeled and can be put together in arbitrary ways, exactly as electronic components in a breadboard. The signals are computed sample by sample such that in any given instant of time the input sample to a component is the result of a delayed version of some other output in a feedback loop. This ideal single sample local memory effectively disassociates the different blocks facilitating the addition and removal of components. With the analog representation utilized in Runge-Kutta, the state space equations are global and removing or adding components signifies rewriting the global system.

We have also developed a novel analog VLSI implementation method for Freeman's model, which can be extended to many other neuromorphic models. One key innovation is to perform time discretization, but to keep the magnitude computations full analog. The resulting hybrid analog-amplitude sampled system is naturally fit for multiplexing. Recall that the most serious limitation to design large Freeman's model in VLSI chips was the size of the interconnect, and multiplexing naturally reduces the size of the interconnect from $O(N^2)$ to $O(N)$.

The VLSI implementation and the digital simulator share the same formal system representation. Then the analog system's lack of flexibility can be compensated by the high level digital system representation. System optimization can be done at this level and the parameters brought into the analog system when finished. Eventually, an integrated development system with a computer and the analog VLSI front-end is possible.

Implementing the dynamic portion of the Freeman model posed a difficult challenge. Long time constants and reduced area are difficult to conciliate and area needs to be spared to maximize the number of channels. Power is also an important variable in this equation. Different possibilities were analyzed namely switch-capacitor (SC) and switch-current (SI). None comply fully with the requirements. Switch-capacitor circuits take too much area and do not conform with ultra low power demands. The SI is preferred because it uses current signals for dynamics and algebraic operations are better attained using current mode circuits. But the gamma kernel implementation displays offset dependence over the feedback parameter. The SI proposed offset compensation scheme has been shown to be very effective. Nevertheless, long time constants still present strong difficulties to these methods. The requirements for large capacitor or transistor ratios limits their usefulness for the Freeman model.

The F&H implementation solves much of the requirements. By sampling the internal states of a continuous-time analog filter, it is possible to extend the impulse response of the filter without changing the system's passive elements. The effect is an equivalent capacitor increased by the sampling clock duty-cycle. A fairly small and ultra-low power second order F&H filter was designed making the analog VLSI implementation more conducive to implement larger Freeman models to attack realistic problems.

Measurements over the KII VLSI cells, demonstrated that the responses were very similar to the digital simulator solution, establishing the high level formal equivalence among both domains. The multiplexing methods were proven effective showing that, in computational terms, multiplexing is a transparent method.

The analysis of the computational properties of the Freeman model is just being initiated. Experimentation is a fundamental task to probe theoretical aspects and test hypothesis. We have created a valuable framework for this inquiry which will help find and formulate the essential computation paradigm used in Freeman's model.

6.2 Future Directions

Much remains to be done in understanding Freeman's olfactory model as an information processing system. The capacity, speed of recall, robustness and processing power of the model needs to be quantified. Interpretation of the network signals is also extremely important. The internal representations need to be further investigated, and, before anything, an optimal readout is fundamental to accomplish meaningful functions. Future research will have to concentrate on the functional behavior for information processing.

As of recently, the model has been used for memory association but, model robustness and accuracy needs assessment, and the results should be validated with other known systems. The DSP simulator plays an important role here because it is a flexible test bed for experiments and probing of the different theories and theses that will grow as the understanding matures about the model. The dynamical behavior of the Freeman model depends on the system parameters, and thus a system parameter refinement of the model should be performed using optimization methods such as backpropagation [68].

We believe we have specified and built the platforms that will help answer some of these questions. Nonetheless some other design problems need to be addressed. As the number of cells increase, the digital control grows. It is necessary to determine if the digital control should share the same chip die with the analog circuitry, or otherwise should be generated externally on a different chip. The last choice seems more promising for two reasons. First, with less digital components sharing a common substrate, less digital noise will be coupled to the analog circuits. Secondly, more space is available for extra channels. But this approach raises the problem of the number of input lines. A selecting line per each channel is impractical, so a coded address line, maybe multiplexed, is needed to save chip pins. This will require a communication protocol, that can also eventually lead to multichip set implementations.

The spatial averaging that occurs at every cell of a network needs also to be addressed. If the number of channels increase, the averaging gains drop dramatically which may cause implementation limitations. Finally, different active resistive loads [69] have to be investigated to allow continuous changes on the system parameters. A critical point element that affects the performance of KII (or KIII) network is nonlinearity which is responsible for most of the differences between cells. It is difficult to physically match them. However if a single nonlinear block is shared among all K0s then we will have uniform behavior and it will simple to set up the system. This can be accomplished with extra multiplexing of this block and adding an extra sample and hold to represent the output of each K0.

APPENDIX A
INTRODUCTION TO SWITCH-CAPACITOR AND SWITCH-CURRENT CIRCUITS, AND DESIGN METHODS

A.1 Introduction

This appendix introduces different discrete-time techniques commonly used in analog VLSI systems. It also introduces the main nonideal effects that degrades the performance and methods commonly reported in literature to minimize those effects.

A.2 Switch-Capacitor—Introduction

Switch-capacitor (SC) [39–41] is probably the most popular technique utilized for filtering. The principle of periodically switching a capacitor charge between two nodes has been known for more than one hundred years [70] as a way to achieve an equivalent resistor. In this sense, any active filter could, in principle, be implemented with this technique by simply substituting each resistor with the equivalent achieved through the capacitor switching. Technological problems kept the technique from realization (until the beginning of the 1970's [71]). Good switches, linear capacitors and very high opamp input impedances are necessary for optimal results. Bipolar only technologies do not fulfill these requisites, especially in terms of the opamps input base currents. However these more prominent aspects can be found in MOS technologies. Other types of filters, such as the continuous-time filters like Gm-C [72] or MOSFET-C [69], have also been widely utilized. More recently, techniques like log domain filters [73,74], exponential state space filters [75] and other current mode filters [76] have also been reported. However, if time constant precision is a concern then SC is the preferred technique. If a filter with time constant $\tau = RC$ (R- for resistor, C- for capacitor) is designed with resistors and capacitors, then a great precision on τ should not be expected. The precision is hindered with the fabrication process of an integrated circuit. Resistors and Capacitors follow different fabrication procedures. This means that the errors associated with yield defects do not track each other making the precision of the product RC very difficult to control. For these reasons, active continuous-time filters in general need tuning (e.g., with automatic procedures [77]) in order to achieve a precise frequency response. In opposition, ratios of capacitors can be made as precise as 0.1% [78]. The capacitor ratios are the time constants in SC filters.

As the "switch" word indicates, SC circuits are discrete-time signal processors (but with analog amplitudes). The SC technology has the necessary scalability, addition, inversion and delay properties that are fundamental and strictly necessary to implement discrete-time linear systems. These individual elements are in general embedded in the topology circuit and, in principle, cannot be found explicitly (such as a delay block). In fact, the most direct way to build an SC filter is by drawing a realization of a continuous-time filter *H*(*s*) and then substituting the resistors with the SC counterparts resulting in an *H*(*z*). If the sample frequency is made infinite then $H(z)|_{z = e^{jwT}} \rightarrow H(s)$.

A direct way to find an SC topology is to find a realization of *H*(*z*) based on the $(z-1)^{-1}$ blocks [79]. This is the more basic memory block and is represented by a simple opamp with a feedback capacitor.

A.2.1 The SC Resistor

The basic principle to achieve an equivalent resistor using charge switching can be easily understood using the classical example of Figure A-1.

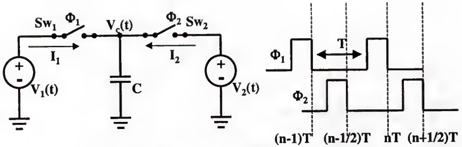


Figure A-1. Switch capacitor resistor

This example has to regard the fundamental assumptions that is made to any SC circuit. It is assumed that the voltages $V_1(t)$ and $V_2(t)$ do not change during the "on" phase of Φ1 and Φ2 respectively. On each phase the respective switch will be closed and connecting the capacitor in parallel with the proper analog voltage source. A non-overlapping two phase clock, Φ1 and Φ2, is needed, and the clock period T is chosen according to the Nyquist criterion [80] (the input voltages,$V_1(t)$ and $V_2(t)$, have to be band limited). The key variable for the analysis is the capacitor (or capacitors) charge variation during the different clock phases. Given the charge $Q=CV_c$ and applying circuit analysis techniques, such as charge conservation principles, the system equation is reduced to a difference equation where the states are node voltages defined at the sampling time intervals:

$$[\ldots,(n-1)T, (n-1/2)T, nT, (n+1/2), \ldots].$$

Consider Figure A-1. The capacitor charge variation when Sw$_1$ closes (ϕ1 is "on") is: $\Delta Q_1 = Q_{final} - Q_{previous} = CV_1 - CV_2 = C(V_1 - V_2)|_{\text{during phase } \Phi 1}$ (assuming that the capacitor was previously charged to V$_2$). The same happens during Φ2 phase: $\Delta Q_2 = Q_{final} - Q_{previous} = CV_2 - CV_1 = C(V_2 - V_1)|_{\text{during phase } \Phi 2} = -\Delta Q_1$. This is repeated every T seconds.

The current, by definition, is the charge variation per time unit or, $\frac{dQ_1}{dt} = I_1 \Leftrightarrow \Delta Q_1 = \int_{(n-1)T}^{nT} I_1 \cdot dt \Leftrightarrow \frac{\Delta Q_1}{T} = \frac{1}{T}\left(\int_{(n-1)T}^{nT} I_1 \cdot dt\right) \Leftrightarrow \frac{\Delta Q_1}{T} = I_{1\text{average}}$. Thus, the capacitor average current is the charge variation per unit sampling time T. Because the voltage sources do not change during the sampling time then:

$$I_{average} = \frac{V_1 - V_2}{T/C} \tag{A-1}$$

$$R_{eq} = \frac{V_1 - V_2}{I_{average}} = T/C \tag{A-2}$$

Considering T/C as an equivalent resistor is valid if $V_1(t)$ and $V_2(t)$ remain constant for all time t or as long as $f_v/f_s \ll 1$ where f_v is the maximum frequency of $V_1(t)$ and $V_2(t)$, and $f_s = 1/T$. This is a high sampling frequency approximation and can always be applied to an SC circuit to find its analog (amplitude and time) counterpart.

In general, Eq. (A-1) does not hold because the over-sampling assumption is not valid. In this situation an $H(z)$ can be defined.

A.2.2 Simple SC Integrator

An integrator, such the one of Figure A-2a, can be used to find the SC counterpart by replacing the resistor R with the equivalent SC resistor realization (Figure A-1) resulting in the integrator of Figure A-2b. The total charge at time $t_1 = (n-1)T$ in both capacitors, C_1 and C_2, referred to the plates connected to virtual ground during Φ2, is: $Q_{t1} = C_1 V_i[(n-1)T] - C_2 V_o[(n-3/2)T] = C_1 V_i[(n-1)T] - C_2 V_o[(n-1)T]$.

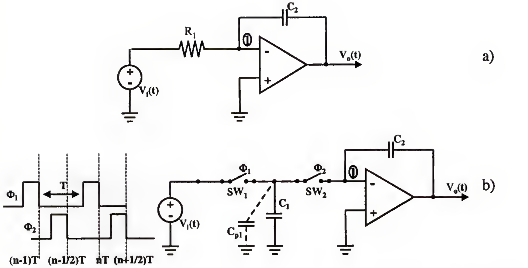


Figure A-2. First order SC integrator. a) Continuous-time integrator; b) Equivalent SC integrator

At time $t2 = (n-1/2)T$ the total charge is $Q_{t2} = -C_2 V_o[(n-1/2)T]$. This charge is only dependent on C_2 because C_1 is shorted between ground and the opamp virtual ground. Because of the non-overlapping clock characteristic and the infinite opamp's input impedance, the total charge Q_{t1} at $t_1 = (n-1)T$ can not be lost at $t_2 = (n-1/2)T$. According to the charge conservation principle then $Q_{t2} = Q_{t1}$. Knowing that $V_o[(n-1/2)T] = V_o(nT)$ then the resulting difference equation is Eq. (A-3) and the correspondent *H*(*z*) is defined by Eq. (A-4).

$$C_1 V_i[(n-1)T] - C_2 V_o[(n-1)T] = -C_2 V_o(nT) \tag{A-3}$$

$$H(Z) = -\frac{C_1}{C_2} \cdot \frac{1}{Z-1} \tag{A-4}$$

If the over-sampling condition is satisfied, then the following expansion of Eq. (A-4) is legitimate:

$$
\begin{aligned}
H(e^{jwT}) = -\frac{C_1}{C_2}\cdot\frac{1}{e^{jwT}-1}\Bigg|_{wT \ll 1} &\rightarrow H(e^{jwT}) = -\frac{C_1}{C_2}\cdot\frac{1}{(1+jwT+\ldots)-1} \cong \\
&\cong -\frac{C_1}{C_2}\cdot\frac{1}{jwT} = \frac{1}{jwC_2R} \\
&R = T/C_1
\end{aligned}
\qquad \text{(A-5)}
$$

Equation (A-5) shows that if the over-sampling condition holds, the continuous-time integrator of Figure A-2a is reasonably approximated by the SC integrator. If not, the system should be characterized with discrete-time tools.

The SC resistor realization shown in Figure A-1 is not unique. There are other possible resistor implementations such as: series, series-parallel, bilinear SC resistors [81]. The switching procedure of Figure A-1 or Figure A-2, although simple, is seldom used because it is sensitive to parasitic capacitors. Any capacitor connected in parallel to a low impedance node or to a virtual ground does not influence the integration constant. However, C_{p1} represented in Figure A-2b, participates directly in the charge sharing because it appears in parallel with C_1 in both clock cycles. In order to eliminate the effect of this capacitor a parasitic insensitive switching [82–84] is needed. The new switch is shown in Figure A-3. During phase Φ1, C_{p1} and C_1 both charge to the voltage V_i. During phase Φ2 the charge of C_1 is transferred to C_2 but not C_{p1} which is shorted to ground by Sw_1. Conversely, C_{p2} is either grounded through Sw_2 or connected to a virtual ground, not participating with any charge to C_2. Thus the integrator of Figure A-3 is insensitive to parasitic capacitors.

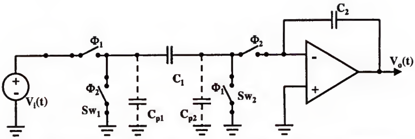


Figure A-3. Non-inverting parasitic insensitive SC integrator

A.2.3 The SC Gamma Filter

Previous sections have briefly introduced the principle and analysis of SC circuits. In this section a gamma filter example is presented. For convenience, the gamma kernel equations, both in the time domain and in the Z domain, are rewritten in Eq. (A-6) and Eq. (A-7).

$$x_k(n) = (1-\mu)\cdot x_k(n-1)+\mu\cdot x_{k-1}(n-1) \tag{A-6}$$

$$G(z) = \frac{X_k(z)}{X_{k-1}(z)} = \frac{\mu}{z-(1-\mu)} \tag{A-7}$$

If $\mu < 1$ Eq. (A-7) represent a lossy integrator. This can be implemented by adding an SC resistor around C_2 (Figure A-3) and shown in Figure A-4

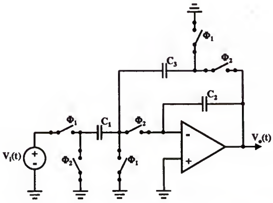


Figure A-4. Lossy integrator (gamma filter with $\mu < 1$)

Applying previous procedures the following difference equation is obtained (making $C_1 = C_3 = C$):

$$\begin{aligned}&(C_2 + C)V_o[nT] = C_2 V_o[(n-1)T] + CV_i[(n-1)T] \Leftrightarrow \\ &\Leftrightarrow V_o[nT] = (1-\mu)V_o[(n-1)T] + \mu V_i[(n-1)T] \Big|_{\mu = \frac{C}{C+C_2}}\end{aligned} \tag{A-8}$$

$$H(Z) = \frac{\frac{C}{C+C_2}}{Z - \left(1 - \frac{C}{C+C_2}\right)} \tag{A-9}$$

The lossy integrator guarantees that the gamma kernels are always stable since $\mu < 1$ as Eq. (A-9) attests. To scale μ, C_2 should be variable.

A more general gamma kernel, where μ can assume any value, can be built as proposed in Figure A-5. The opamp Op_1 and $C_{s/h}$ form a sample and hold. The difference equation and corresponding transfer function are defined in Eq. (A-10) and (A-11), respectively ($C_1 = C_3 = C$).

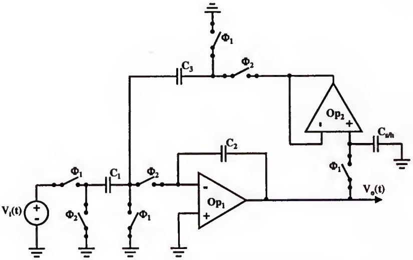


Figure A-5. The SC gamma kernel

$$V_o[nT] = (1-\mu)V_o[(n-1)T] + \mu V_i[(n-1)T] \Big|_{\mu = \frac{C}{C_2}} \tag{A-10}$$

$$H(Z) = \frac{\frac{C}{C_2}}{Z - \left(1 - \frac{C}{C_2}\right)} \tag{A-11}$$

If C_2 is made variable, the μ value can be greater than 1 however in this case care should be taken concerning stability. The circuit of Figure A-5 was designed using ORBIT—MOSIS 2.0μ technology. In order to test the circuit with several time constants, multiplexers were used to program the C_2 and C values [78]. The overall circuit is shown in Figure A-6.

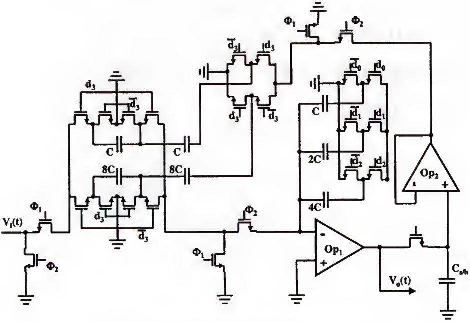


Figure A-6. Programmable gamma kernel

In Figure A-6, d_i and $\bar{d}_i$ i=0..3, are digital voltages assuming the values "0" or "1". These voltages control the multiplexers to program the μ parameter of the gamma filter which can be defined by Eq. (A-12). The capacitor values were chosen to encode seven different values of μ. Since C_1 and C_3 (Figure A-5 and Figure A-6) can assume two different values, another seven μ values are encoded, making a total of fourteen different μ values. Depending if d_3 is "1" or "0", μ will be smaller or greater than 1, respectively.

$$\mu = \frac{2^{(3 \cdot d_3)}}{\left(\sum_{i=0}^{2} [2^i \cdot d_i]\right)} \tag{A-12}$$

The chip was designed with three gamma taps in cascade. Figure A-7 and Figure A-8 shows the measured results. The results were shown to be within 3% precision on the time constant. The offset is about 60mV. The error on the time constant, due to capacitor imprecision, can be observed with μ=8/4, which is the stability limit. The expected result should be a steady oscillation (pole at z=-1), however the measured signal was still high-pass with damped overshoot. For μ>2 the output saturated as expected.

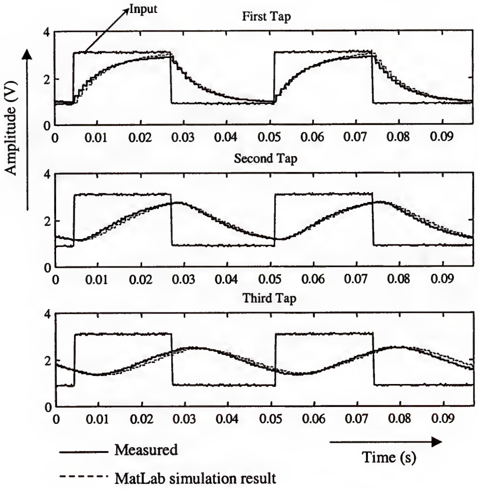


Figure A-7. Gamma filter result with μ=1/7

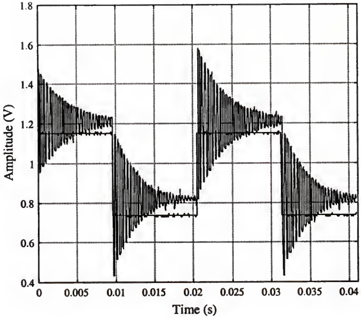


Figure A-8. Measured result for μ=8/4

A.3 Charge Injection (Clock Feedthrough CFT)

The simplest component in sampling systems, the switch, is also the largest source of offset, gain errors, and noise. Howbeit these nonidealities, the MOS (or CMOS) switch is a fundamental piece that helps sampled systems, such as SC, to succeed. The switches commonly utilized in VLSI sampled systems are shown in Figure A-9.

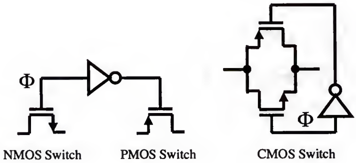


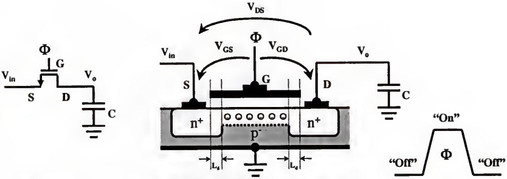
Figure A-9. Different types of MOSFET switches

Between the NMOS and PMOS switches there are no substantial differences. However, they impose a real limitation on dynamic range. If the clock Φ varies between 0 and V_{dd} the maximum signal voltage allowed at the source of the NMOS switch is $V_{dd} - V_{th}$ (with V_{th} the threshold voltage). For the PMOS switch the minimum voltage would be V_{th}. With these switches the dynamic range cannot span the full power supply range and it is further aggravated by the body effect [85] which increases the $|V_{th}|$ value. Since the NMOS has an upper signal limit, and the PMOS limitation is at the lower end of the signal, placing these two switches in parallel (Figure A-9—CMOS switch) the dynamic range will span the power supply. When the NMOS cuts-off at $V_{dd} - V_{th}$, the PMOS will be conducting up to V_{dd}. At the lower end the opposite occurs. When both transistor are conducting the switch equivalent resistor is the parallel of each isolated equivalent value, lowering the effective resistor value of the switch. The drawback is that it needs an inversion of the clock signal Φ and a bit more area which per se is not of great importance when contrasted with the single transistor switch.

Although MOS switches, in general, present good switching solutions, they are not without dynamical shortcomings. The dynamic nonideal effects associated with MOS switches eventually becomes the cause for signal to noise ratio (S/N) deterioration, offsets, gain errors and distortion. Ultimately the effect responsible for these specification limits is charge injection also called clock feedthrough (CFT) [55–57,40].

A.3.1 Charge Injection

The best way to understand the nonideal phenomenon of charge injection is to consider a simple sample and hold (S&H) shown in Figure A-10.



S&H Schematic

S&H with the MOSFET physical representation (transversal cut)

Figure A-10. Sample and hold (S&H) circuit (physical representation when Φ="on")

When the Φ clock signal is "on", the transistor accumulates inversion charge under the gate allowing the capacitor to charge up to the input voltage V_{in}. Since it is assumed that the "on" state of the clock is at a voltage value such that $V_{GS} > V_{th}$, then the MOSFET transistor will be in the linear region ($V_{DS} < (V_{GS} - V_{th})$) because $V_{DS} \to 0$. In this situation the transistor charge distributes "uniformly" along the channel from the source (S) up to the drain (D).

The error occurs when the transistor goes into the "off" state (when Φ transits from "on" to "off" state). When Φ goes to "off" state, the charge in the channel under the gate has to disappear, withholding any current from flowing and the transistor is "off" (assuming zero charge for $V_{GS} < V_{th}$, which is not true but a good approximation under the considerations). This charge will obviously be deviated or injected into the drain and source circuit paths, causing then the charge injection.

The injected charge into V_{in} (Figure A-10) does not generate errors because V_{in} is assumed to be a low impedance driving source and its value will not change. However, such will not be the case on the capacitor side. The extra charge coming from the transistor channel will be algebraically added to the charge already there, changing the stored voltage. Under this circumstance the equality $V_o = V_{in}$ is no longer valid and a sampling error occurs. If the clock signal transit states fast enough, it can be assumed that half the channel charge is injected into the capacitor. The voltage change due to this charge can be quantified as Eq. (A-13) [81].

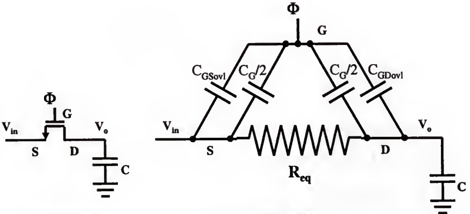
$$\Delta V_o = -\frac{C_{ox} \cdot WL \cdot (V_{DD} - V_{in} - [V_{th0} + \gamma(\sqrt{|2\Phi_F| + V_{in}} - \sqrt{|2\Phi_F|})])}{2C} \qquad \text{(A-13)}$$

Although the process and modeling of the charge injection is more complex than it is apparent from Eq. (A-13), it illustrates well the detraction effect. The more important aspect of the charge injection is the nonlinear dependence (square root) of the error voltage with the input signal V_{in}. Inevitable harmonic distortion will result.

A.3.2 Clock Feedthrough from Overlap Capacitive Coupling

A more correct model can be derived for the dynamical nonidealities using the schematic of Figure A-11[56]. Although we are not going to derive the exact model [56], it illustrates well the CFT process. During the drain and source area etching and implanting fabrication process, some of the gate will overlap into these areas with a length extent denominated by lateral diffusion length (L_D in Figure A-10). The overlap can be model as a capacitive coupling between the gate, drain and source with a value defined by: $C_{GSovl} = C_{GDovl} = W \cdot L_D \cdot C_{ox}$. The capacitor C_G (Figure A-10) represents the gate capacitor ($C_G = W \cdot L \cdot C_{ox}$). With the transistor in the "off" state there is no more charge in the channel however, due to C_{GSovl} and C_{GDovl}, there will still be charge change in C during the transition period when Φ goes from $V_{in} + V_{th}$ to zero Volt. The charge injection caused by the overlap capacitors before the transistor cuts-off, is negligible because the gate capacitor is much bigger than the overlap capacitors. Nevertheless, they are the elements solely responsible for clock coupling when the control voltage goes below the threshold voltage. The equivalent model for CFT when $\Phi < (V_{in} + V_{th})$ is represented in Figure A-12 [56]. If the clock transition is very fast, the error voltage due to the overlap capacitors can be defined by Eq. (A-14).

$$\Delta V_o = \frac{C_{GDovl}}{C_{GDovl} + C} \cdot V_{DD} \qquad \text{(A-14)}$$



S&H Schematic

S&H Model

Figure A-11. S&H model for clock feedthrough [56]

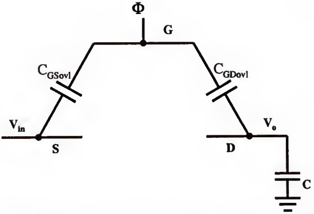


Figure A-12. The S&H model when $\Phi < V_{in} + V_{th}$

The error voltage ΔV_o is approximately independent to the input voltage. In this case the overlap capacitors will only be accountable for offsets.

A.3.3 Control over CFT

Control over CFT is very limited however, there are trade-offs and a few circuit solutions that the designer should be aware in order find the best possible compromise. According to the above exposition and with the help of equations (A-13) and (A-14) the following summarized points should be considered:

- ΔV_o increases with transistor area. This is evident from Eq. (A-13) and from the actual value of C_{GDovl}. As the area increases so increases C_G which will in turn raise the inversion charge. The area as to be a trade-off between "on" resistance and CFT error. In general the operation frequency will dictate the highest "on" equivalent resistor.
- ΔV_o decreases with the increase of the sampling capacitor value. The control of ΔV_o through C value has implications on chip area and on the time it takes to charge the capacitor up to V_{in}.
- ΔV_o depends on the input voltage. The signal dependence is nonlinear as evidenced in Eq. (A-13) resulting in harmonic distortion. Lowering V_{in} will diminish harmonic distortion but has implications on dynamic range and S/N.
- The slower the transitions of the clock Φ the lower is the CFT. This is not apparent from above the explanations, however from the model of Figure A-11 the statement is made clearer. In this model the charge change in the channel of the transistor is modeled by a nonlinear resistor, R_{eq}, dependent on the controlling voltage Φ [56]. In this sense, if the transition is made slow some of the error charge injected into C can be retrieved by V_{in} through R_{eq}.
- If V_{in} is not an ideal voltage source then the CFT increases with its internal resistance R_s. This point is directly related with the last. This resistor will come in series with R_{eq} increasing the time constant.

Circuit procedures to reduce CFT are limited. The dummy switch [86,87] is a classic solution often used (Figure A-13). When Φ switches to "on" state, M2 will be cutting off and its channel charge is retrieved by V_{in} through M1. However when Φ switches to "off" half of M1 channel charge will be injected into C. Yet M2 is turning "on" and since its size is half of M1s that charge is going to be absorbed by M2 gate capacitor. Thus ideally no charge injection affects the capacitor C. In reality transistor mismatch will limit greatly the configuration efficiency.

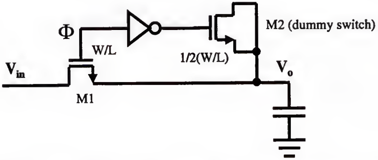


Figure A-13. Cancellation of dynamical switch errors with dummy switch

CMOS switches can also be used for CFT cancellation (Figure A-9). Due to the complementary clock characteristic and knowing that PMOS and NMOS transistors have channel charges of opposite sign, in principle charge injection would be eliminated. However cancellation with CMOS switches demands for precise control over the complementary clocks [81] and symmetry is difficult to attain.

The best solution is to use full differential topologies [88]. An example of such a structure is found in Figure A-14 representing the same integrator in Figure A-3. The transfer function is unaltered but at least the common mode signals at the input of the full differential amplifier will be attenuated. However the signal dependent nonidealities that do not represent common mode signals will eventually limit the linearity and noise performance. An obvious disadvantage with full differential topologies is the need for double resources and consequently, area, complexity, and eventually power degradation. Nonetheless, the differential nature is accountable for doubling the dynamic range, and for better distortion performance [72,81]. Additionally, by simply changing the voltage relative references, an inverted version of the signal is obtained avoiding the need for extra circuit inverting procedures.

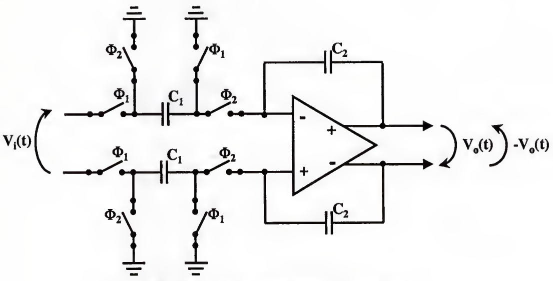


Figure A-14. Full differential SC integrator

1.4 Technological Evolution and its Implications on Analog Design

The continuous pressure for more integration of bigger and more complex systems requires simpler circuit content, with greater area efficiency and technology down scale for miniaturization. Technological feature sizes down scaling has strong implications on device parameters. This condition, combined with the fact that these technologies are mostly optimized for digital operation, makes the implementation of analog functions considerably difficult. At submicron sizes, the short channel effect is very strong, making the analyses more difficult. This does not represent a real problem for analog design because the transistors can be made bigger. However, miniaturization has stronger implications at other levels. For example, when the length of a transistor is made too small, punchthrough effect can occur [85]. The phenomenon can be observed as a strong increase of current as the V_{DS} (Drain Source voltage) increases, which is a direct consequence of source and drain depletion regions closeness in a short length condition. To circumvent the punchthrough problem these depletion regions should be made smaller. It can be achieved by decreasing reverse bias voltages and increasing substrate doping concentration [85]. The bias condition implies lower power supply voltages. The increased doping increases threshold voltage, which may be corrected by making the gate insulation oxide thinner.

Punchthrough is not the only factor to condition the technology parameters. Impact ionization, which is more severe for small devices, and hot carrier effects are also conditioning factors. Especially, the latter degrades the device performance permanently and results from accumulation of charge in the oxide. The charge gets trapped and never leaves (even when the device is powered off) with adverse consequences (e.g. threshold voltage increase). This charge is injected from high energy carriers (either holes or electrons) called *hot carriers* generated from high electric fields and are directly related with the bias voltages (V_{GS} - Gate Source voltage and V_{DS}). All these phenomena are observed above a given “threshold” or breakdown voltage, implying limitations on devices operating voltages. Futhermore with the increase digital scale of integration and the consequent increase in power, it further pushes for low-voltage technologies.

Thus it is clear that the constant miniaturization to accommodate a need for bigger integrated systems has strong implications on device technological parameters and voltage operations. The implications on analog design are now evident. Dynamic range is lost and consequently the signal to noise ratio (S/N) degrades, making the design of analog functions with high S/N and high speed together very difficult (e.g., making the design of the cascode cells difficult). As the technology scales further down eventually the threshold voltage has to be scaled for speed requirements [89] and obviously optimized for digital performance, which may prove to be another extra burden for analog designers.

Nevertheless, no matter how big a digital system is, there will be always a portion of it that has to be analog at the interface level with the outside analog world. Due to the technological evolution mentioned earlier, it means that new methods of analog design have to be developed to resolve expected difficulties. There are some clever applications where analog memory cells [90] are utilized to level shift the threshold voltage [91], or the current mode methods.

1.5 Current Mode Circuits—a Possible Solution

Switch-current circuits proposed by Hughes et al, [42] are a possible method to avoid low voltage problems through operation in the current domain. The idea is based again on the fact that the low voltage operation requirements makes the design of voltage mode circuits, with high linearity and dynamic range, rather difficult. To avoid large voltage swings a low impedance analog circuit is used with current as the signal instead of the voltage. These can operate at lower power supplies because the current flows through low impedance nodes making the voltage swing small [43]. This factor also has implications in frequency since lower impedance in the signal path means a wider frequency band.

Other current mode for general purpose applications (as opamps) have also been reported. Such cells are denominated current-conveyor by Sedra and Smith [92–94] and can be used to perform useful signal processing applications [43].

1.6 Switch-Current Filter

In the linear discrete-time signal processing sense, switch-current (SI) circuits is an universal technique. It finds applications in analog to digital converters [95] and also reports on SI multipliers [96] are available. Implementations either in CMOS or BiCMOS technologies have also been reported [97].

Any difference equation can, in general, be implemented with this technique. The fundamental computational ingredients such as the delay block (memory), summation, inversion, and scaling, are available. The SI sampling blocks have similarities with the current copier circuits [98,99] and was proposed originally by Hughes et al [42]. The basic principle of a SI circuit is very simple. Consider the current mirror amplifier of Figure A-15. It complies with the requisites of a low input impedance and large output impedance current amplifier.

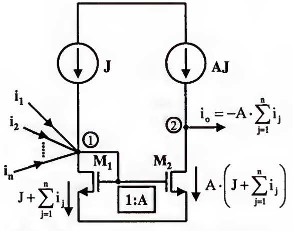

J
AJ
i_1
i_2
i_n
①
②
$i_o = -A \cdot \sum_{j=1}^{n} i_j$
M_1
M_2
1:A
$J + \sum_{j=1}^{n} i_j$
$A \cdot \left(J + \sum_{j=1}^{n} i_j\right)$

Figure A-15. Scaled 1:A current amplifier

Node 1 in Figure A-15, represents the input summing node. The summed current flows through transistor M_1, is mirrored into M_2 and scaled by a factor A. Finally the output current is $i_o = AJ - A\left(J + \sum_{j=1}^{n} i_j\right) = -A \cdot \sum_{j=1}^{n} i_j$. Thus, the current amplifier of Figure A-15 accomplishes the summation, scaling and inversion needed for linear discrete-time signal processing. The delay can be accomplished with the switching procedure of Figure A-16.

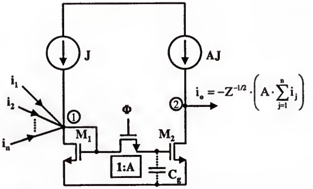

J
AJ
i_1
i_2
i_n
①
②
Φ
$i_o = -Z^{-1/2} \cdot \left(A \cdot \sum_{j=1}^{n} i_j\right)$
M_1
M_2
1:A
C_g

Figure A-16. The SI track and hold circuit

The memory operation is performed by sampling the gate source voltage of M_2 taking advantage of the unavoidable MOS gate capacitor (seen dashed in Figure A-16) to hold that voltage. To understand better the sampling procedure take in consideration Figure A-17 [100].

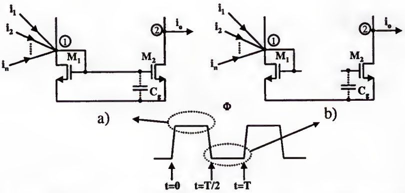


Figure A-17. The SI track and hold operation. a) Track phase; b) Hold phase

When the clock signal Φ goes into "on" state at $t = 0(s)$, the NMOS switch is in the linear region shorting together the gates of M_1 and M_2 (Figure A-17a). During this phase $(0 \leq t < T/2)$ the SI memory cell is working as a current amplifier (Figure A-15) where M_2 equals M_1's current (apart from a possible scale factor if the gain needs to be other than 1). The output current is tracking the input in this phase.

The NMOS switch cuts-off when the Φ transits into an "off" state. The gate capacitor C_g will be holding the voltage it had just before the switch opened at $t = T/2^{-}(s)$, holding the output current. Thus, during the period $T/2 \leq t < T$ (Figure A-17b), denominated holding period, the output current is constant and equal to the input at $t = T/2^{-}(s)$. The complete cycle can then be formalized according to Eq. (A-15). With the added switch in Figure A-16, the four basic processing operations are then realized: summing, scaling, inversion and memory, $z^{-1/2}$.

$$\begin{cases} i_o(t) = -A \cdot \sum\limits_{j=1} i_j(t); & 0 \leq t < T/2 \\ i_o(t) = -A \cdot \sum\limits_{j=1}^{n} i_j(T/2); & T/2 \leq t < T \end{cases} \tag{A-15}$$

There are basically two practical occurrences that limit the SI accuracy, the device mismatch and clock feedthrough. The next sections will try to expose some the sources and possible "solutions".

A.6.1 Device Mismatch

Consider the equation of a NMOS transistor working in strong inversion. The drain/source current is defined by Eq. (A-16).

$$I_{ds} = \frac{\beta}{2} \cdot (V_{gs} - V_t)^2 \times (1 + \lambda V_{ds}) \qquad \beta = K_N \cdot \frac{W}{L} \qquad \text{(A-16)}$$

Consider for reference Figure A-15. This figure represents the T/H (track and hold) cell of Figure A-16 when the switch is "on". This is a critical state because a transference of current from input to the output with minimal error is intended. In an ideal situation the following equalities have to hold in order for the error to be minimal (the subscript numbers 1 and 2 correspond to transistors M_1 and M_2, respectively) [101]:

1. Device aspect ratios: $A \cdot \frac{W_1}{L_1} = \frac{W_2}{L_2}$
2. Threshold voltages: $V_{t1} = V_{t2}$
3. Transconductance parameter: $K_{N1} = K_{N2}$
4. Channel length modulation parameter: $\lambda_1 = \lambda_2$
5. Drain source voltage: $V_{ds1} = V_{ds2}$

Violations of the above equalities will bring gain distortion and DC-offset errors [102]. Points 1 and 5 are variables that the designer can manipulate. The other variables in points 2, 3, and 4 are related with the technology which are premises for the designer. All points are assumed as true for the designer, especially 2 through 4, however in practice during the fabrication process of an integrated circuit, uniformity of parameters at different points of the chip is not expected. In fact due to fabrication variations [101,103], any of the five points presented will ever be matched in space. Devices in different locations will have different technological and physical parameters. What the designer can and should do is try better designs that may attenuate the effects by making the circuits more independent from the largest sources of mismatches.

A.6.2 Mismatch Reduction in CAD Design

During the CAD design some care must be taken to reduce the mismatches mentioned above. An example is to use common centroid geometries [39]. The transistors to be matched are partitioned into several transistors in parallel. They are then cross distributed around the same area putting the transistors in the same space variation conditions (Figure A-18).

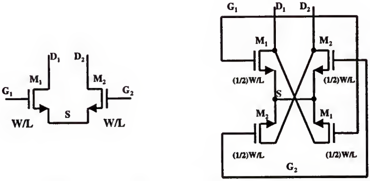


Figure A-18. Schematic of a common centroid geometry for a differential pair

The disadvantage with the common centroid design is the extra area required by the splitting procedure. It requires the transistors to be placed further apart which also is a condition to avoid when possible. A better approach would be the use of stacked layout [104]. A schematic example is shown in Figure A-19 for a current mirror.

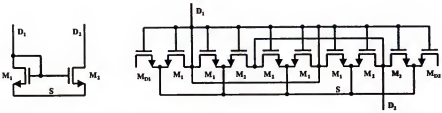


Figure A-19. Schematic of a stacked layout for a current mirror (each transistor in the stack is 1/4 of the original isolated transistor size W/L)

With the stacked layout procedure, unitary transistors are connected in parallel. To improve the matching, the unitary transistors are interdigitized to equally distribute the spacial parameters variation among the equivalent transistor pair. This technique can further be combined with common centroid geometry.

Splitting a transistor into unitary pieces has further benefits. It makes the aspect ratio of the transistor more manageable for layout, it reduces the total parasitic capacitance due to drain and source implant areas, and it improves the ratio sizes precision of a transistor pair [104,64] ($A = (W_2/L_2)/(W_1/L_1)$ in Figure A-15). The same is achieved with linear capacitor design. However, it is not expect similar matching accuracy because more steps are required to build transistors.

The relative position of the matching transistors is also important. If, for example, one of the transistors is rotated 90^o with respect to the other, as much as 2% current mismatch results [101]. In general, the geometric position of the sources and drains relative to each other is important [44]. Figure A-19 shows that along the stack the drains and sources alternate in orientation. For this reason, when the stacked layout is used, the number of split transistors should be even [44]. With even number of transistors, half of the drains(sources) will be to one side and the other half to the opposite side, equalizing the effects.

During the fabrication process, the etching procedure, that defines the gate, overetches the thin oxide under the polysilicon gate material. This process is boundary dependent [104]. Because the transistors on the periphery and the innermost transistors of the stack are affected differently, mismatch occurs. However, the mismatch can be attenuated by putting dummy transistors at both ends of the stack to equalize the boundary geometry (Figure A-19, M_{D1} and M_{D2}).

A.7 Reducing Mismatch Electronically

Simple mathematical manipulations to better understand the influence of the diverse matching variables present in all VLSI designs, can be performed [100,102], giving much insight into the matching problem. Having in mind Figure A-15, the gate source voltage of M_1 is Eq. (A-17) and the output current is going to be given by Eq. (A-18). With $V_{gs2} = V_{gs1}$ substituting Eq. (A-17) into Eq. (A-18) results in Eq. (A-19).

$$V_{gs1} = V_{t1} + \sqrt{\left(\frac{2(J+i)}{\beta_1}\right)/(1+\lambda V_{ds1})} \qquad ;i = \sum_{j=1}^{n} i_j \tag{A-17}$$

$$i_o = A \cdot J - I_{ds2} = A \cdot J - \frac{A \cdot \beta_2}{2}(V_{gs2} - V_{t2})^2(1+\lambda V_{ds2}) \tag{A-18}$$

$$\begin{gathered} i_o = -i \cdot A \cdot \frac{\beta_2}{\beta_1} \cdot \frac{1+\lambda_2 V_{ds2}}{1+\lambda_1 V_{ds1}} - \\ \left\{J \cdot A \cdot \left(1 - \frac{1+\lambda_2 V_{ds2}}{1+\lambda_1 V_{ds1}} \cdot \frac{\beta_2}{\beta_1}\right) + (1+\lambda_2 V_{ds2}) \cdot A \cdot \frac{\beta_2}{2} \Delta V_t^2\right\} - \\ \Delta V_t \cdot gm_2 \cdot \left(\sqrt{A \cdot \frac{\beta_2}{\beta_1}} \cdot \frac{1+\lambda_2 V_{ds2}}{\sqrt{1+\lambda_1 V_{ds1}}}\right) \cdot \sqrt{1+\frac{i}{J}} \\ \begin{Bmatrix} gm_2 = \sqrt{2A\beta_2 J} \\ \Delta V_t = V_{t1} - V_{t2} \end{Bmatrix} \end{gathered} \tag{A-19}$$

In an ideal situation, Eq. (A-19) should be reduced to $i_o = -A \cdot i = -A \cdot \sum_{j=1}^{n} i_j$. But instead, harmonic distortion will occur due to the nonlinear square-root dependence of the output with the input current (third term on right hand side of Eq. (A-19)). Considering that the input signal is small enough to neglect the Early effect λ, the technological parameter responsible for the harmonic distortion is the threshold voltage mismatch $\Delta V_t = V_{t1} - V_{t2}$. It is obvious that as the ratio i/J gets smaller then smaller is the effect of the square-root in the overall equation. So if the biasing current J is increased, the THD (total harmonic distortion) and gain error will decrease [105]. The other technological parameters will only cause DC offset. The increase of the biasing current J in relation to the signal current i, will further make the variation of V_{ds} less dependent on the output signal current. However, making i/J small will obviously inflict a penalty on the dynamic range and power.

A new configuration, which greatly improves the matching properties of the track and hold (T/H) circuit (Figure A-16), was proposed by Hughes et al [106]. It is named "Second generation Switch-current" cells. The method is based on the current copier [98,99] and the dynamic current mirror [107] and virtually eliminates the matching problem inherent to the T/H circuit of Figure A-16. The new cell is shown in Figure A-20

The operation is as follows. When S_1 and S_2 closes, M_1 is diode connected and offers a low impedance path for the input current, that charges the capacitor C_g with the necessary V_{gs} voltage for that current $(J + i_i)$. When these switches open and S3 closes, M_1 will be a current source given by Eq. (A-16) where V_{gs} is the voltage previously stored in C_g, so $i_o = z^{-1/2} i_i$. Since the current sink and current source is performed with the same transistor, the mismatch problems discussed in Section A.6.1 are virtually inexistent. Nonetheless, when scaling is needed or the current needs to be duplicated, a mirror is necessary. Thus, the memory cell is improved with the second generation configuration however, the scaling will eventually introduce all the problems discussed in Section A.6.1.

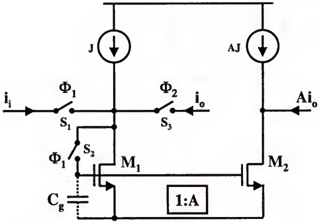


Figure A-20. Second generation SI cell

A.7.1 CFT in SI Circuits

The CFT in SI circuits is in all similar to SC circuits. Thus, all considerations made in Section A.3.3 to control the CFT can be directly applied to the SI circuits. However, one simple rule should be added: making the DC V_{gs} large (by either increasing the bias current J or decreasing W/L) reduces CFT. The V_{gs} increase also contributes to improve the THD and gain errors due to threshold voltage mismatch.

Some interesting strategies have been proposed to cancel the CFT. One of the first circuits proposed to cancel the signal independent (offset) CFT is represented in Figure A-21 [102]. A dummy memory cell generates a bias current corrupted with CFT component. This current is then subtracted to the actual output signal current. Thus, the resulting output current will be exempt of signal independent CFT. This method has a very limited applicability because it relies on the matching between the two cells, which is difficult to realize.

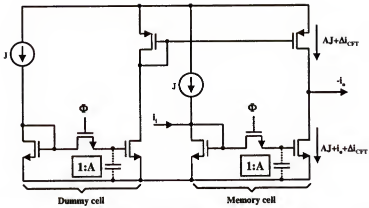


Figure A-21. Clock feedthrough cancellation using current replica

A similar technique that is able to cancel the signal dependent CFT to some extent, is shown in Figure A-22 [108]. The input signal is memorized in two identical cells but with size ratios of 1:2 (if the ratio were 1:1 the added signal at the output would be zero). When the cell currents are added together, and if the CFT current is the same in both cells, the CFT will be cancelled. However, this method suffers from the very same matching problem discussed for the current replica cancellation technique.

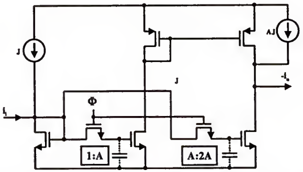


Figure A-22. Signal dependent CFT cancellation using the current replica technique

The most utilized technique to cancel the CFT with single ended signals is the so called S^2I switch-current cells [109] (or the more general n-step charge injection cancellation [110]). The S^2I switch-current is illustrated in Figure A-23 for an S^2I half clock ($z^{-1/2}$) memory cell.

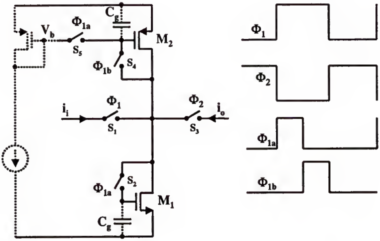


Figure A-23. The S^2I half clock memory cell

The memory is a two step procedure, namely the coarse and fine steps [109]. The primary principle is that during the phase clock Φ_{1a} transistor M_1 is "memorizing" the input current and M_2 is sourcing the bias current J (S_1, S_2 and S_5 are closed). During Φ_{1a} phase clock, the gate capacitor of M_2 will be holding a memory of the bias voltage V_b. This is named the coarse memory period. When the switch S_2 and S_5 opens, a current $I_1 = J + i + \Delta i$ will be flowing through M_1, where Δi represents the CFT injected noise when the switches open. The fine step is during Φ_{1b}. At this stage, the voltage at the gate of M_2 has to change to equalize the extra current Δi. Thus, the current at M_2 will now be $I_2 = J + \Delta i$. At the end of the fine phase all switches are open except S_3 that will be closing. If no CFT were injected by opening S_4, the CFT error to the output would be zero and independent of the input signal. In reality it will not be zero but much less than any other technique presented so far. The only CFT in the output will be caused by S_4 with a signal dependence on Δi that should be $\Delta i \ll (i, J)$. Consequently this technique is almost free from CFT errors.

Still, the best procedure to reduce the errors for sampled-data systems, is the fully differential circuits. Examples of such cells can be found in [111–113] or more recently [114].

A.7.2 Finite Input and Output Impedance

One important cause for errors, not yet mentioned, is the input(output) and output(input) conductance(impedance) ratio. If an SI cell has a high input impedance and a low output impedance, a good input/output current transfer is not expected when two $z^{-1/2}$ cells are cascaded. For an optimal current drive and sink, the input conductance should be as high as possible and the output close to zero. An increase of the input conductance can be achieved using voltage amplifiers to increase the loop gain of the diode connected transistors [113,115]. This technique is often used in amplifiers to decrease the output stage open-loop impedance [116]. The method is efficient but voltage feedback defeats the purpose of low voltage and large bandwidth operation in current mode circuits. A different option is to increase the output impedance (decrease g_{out}) and techniques such as the cascode and high-swing cascode [42] are the most common methods.

APPENDIX B
THE SI LOCAL OFFSET CLOSE FORM OUTPUT EQUATION

This appendix presents the close form derivation for the SI gamma local offset canceling scheme of Chapter 3. For convenience equations (3-7) and (3-8) are rewritten here in Eq. (B.1) and Eq. (B.2).

$$O_1(n) = \begin{cases} X_{of1} & ,n \ even \\ (1-\mu)\cdot[O_2(n-1)-O_1(n-1)]+X_{of1} & ,n \ odd \end{cases} \tag{B.1}$$

$$O_2(n) = \begin{cases} (1-\mu)\cdot[O_1(n-1)-O_2(n-1)]+X_{of2} & ,n \ even \\ X_{of2} & ,n \ odd \end{cases} \tag{B.2}$$

To find a close form solution for these equation a few iterations have to be performed over Eq. (B.1) and Eq. (B.2). From these a close form can be intuited. With the result a proof by induction is sufficient to attest the validity of the close form.

Assume that the initial conditions are $O_1(-1) = X_1$, $O_2(-1) = X_2$ and that the time starts at $n = 0$. According to equations (B.1) and (B.2), the iterative Eq. (B.3) can be found. The close form equations have to be considered for the sequences n odd and n even, consequently two hypothesis have to be tested, for n even and n odd. After some minor manipulations, the following thesis is proposed.

$$n = 0 = even \rightarrow \begin{cases} O_1(0) = X_{of1} \\ O_2(0) = (1-\mu)\cdot(X_1 - X_2) + X_{of2} \end{cases}$$

$$n = 1 = odd \rightarrow \begin{cases} O_1(1) = (1-\mu)^2\cdot(X_1 - X_2) + (1-\mu)\cdot X_{of2} - (1-\mu)\cdot X_{of1} + X_{of1} \\ O_2(1) = X_{of2} \end{cases}$$

$$n = 2 = even \rightarrow \begin{cases} O_1(2) = X_{of1} \\ O_2(2) = (1-\mu)^3\cdot(X_1 - X_2) + (1-\mu)^2\cdot X_{of2} - (1-\mu)^2\cdot X_{of1} + (1-\mu)\cdot X_{of1} - \\ (1-\mu)\cdot X_{of2} + X_{of2} \end{cases}$$

$$n = 3 = odd \rightarrow \begin{cases} O_1(3) = (1-\mu)^4\cdot(X_1 - X_2) + (1-\mu)^3\cdot X_{of2} - (1-\mu)^3\cdot X_{of1} + (1-\mu)^2\cdot X_{of1} - \\ (1-\mu)^2\cdot X_{of2} + (1-\mu)\cdot X_{of2} - (1-\mu)\cdot X_{of1} + X_{of1} \\ O_2(3) = X_{of2} \end{cases} \tag{B.3}$$

$$n = 4 = even \rightarrow \begin{cases} O_1(4) = X_{of1} \\ O_2(4) = (1-\mu)^5\cdot(X_1 - X_2) + (1-\mu)^4\cdot X_{of2} - (1-\mu)^4\cdot X_{of1} + (1-\mu)^3\cdot X_{of1} - \\ (1-\mu)^3\cdot X_{of2} + (1-\mu)^2\cdot X_{of2} - (1-\mu)^2\cdot X_{of1} + (1-\mu)\cdot X_{of1} - (1-\mu)\cdot X_{of2} + X_{of2} \end{cases}$$

...

B.1 Close form proof by induction

B.1.1 Thesis

The close form of Eq. (B.1) and Eq. (B.2) is represented by Eq. (B.4) and Eq. (B.5).

$$n = even \rightarrow \begin{cases} O_1(n) = X_{of1} \\ O_2(n) = (1-\mu)^{n+1}\cdot(X_1 - X_2) + \left(\sum_{k=0}^{n}[(1-\mu)^k\cdot(-1)^k]\right)\cdot X_{of2} + \\ \left(-\sum_{k=0}^{n}[(1-\mu)^k\cdot(-1)^k] + 1\right)\cdot X_{of1} \end{cases} \tag{B.4}$$

$$n = odd \rightarrow \begin{cases} O_1(n) = (1-\mu)^{n+1}\cdot(X_1 - X_2) + \left(-\sum_{k=0}^{n}[(1-\mu)^k\cdot(-1)^k] + 1\right)\cdot X_{of2} + \\ \left(\sum_{k=0}^{n}[(1-\mu)^k\cdot(-1)^k]\right)\cdot X_{of1} \\ O_2(n) = X_{of2} \end{cases} \tag{B.5}$$

B.1.2 Hypothesis

If n is even then Eq. (B.6) is observed.

$$n+1 = odd \rightarrow \begin{cases} O_1(n+1) = (1-\mu)^{n+2}\cdot(X_1-X_2)+\left(-\sum_{k=0}^{n+1}[(1-\mu)^k\cdot(-1)^k]+1\right)\cdot X_{of2}+ \\ \left(\sum_{k=0}^{n+1}[(1-\mu)^k\cdot(-1)^k]\right)\cdot X_{of1} \\ O_2(n) = X_{of2} \end{cases} \qquad (B.6)$$

Otherwise, if n is odd Eq. (B.7) is observed.

$$n+1 = even \rightarrow \begin{cases} O_1(n+1) = X_{of1} \\ O_2(n+1) = (1-\mu)^{n+2}\cdot(X_1-X_2)+\left(\sum_{k=0}^{n+1}[(1-\mu)^k\cdot(-1)^k]\right)\cdot X_{of2}+ \\ \left(-\sum_{k=0}^{n+1}[(1-\mu)^k\cdot(-1)^k]+1\right)\cdot X_{of1} \end{cases} \qquad (B.7)$$

B.1.3 Proof of the hypothesis veracity

- The first elements of the series are $n = 0$ and $n = 1$. Taking these values into equations (B.4) and (B.5), respectively, results in:

$$\begin{cases} O_1(0) = X_{of1} \\ O_2(0) = (1-\mu)\cdot(X_1-X_2)+X_{of2} \end{cases} \qquad (B.8)$$

$$\begin{cases} O_1(1) = (1-\mu)^2\cdot(X_1-X_2)+(1-\mu)\cdot X_{of2}+X_{of1}-(1-\mu)\cdot X_{of1} \\ O_2(1) = X_{of2} \end{cases} \qquad (B.9)$$

This result matches the one for $n = 0$ and $n = 1$ in Eq. (B.4).

∴ Equations (B.4) and (B.5) are representations for the first element of (B.1) and

- If n is even then $n+1$ is odd. Taking Eq. (B.4) and substituting in Eq. (B.1) and Eq. (B.2) for n odd results in:

$$\left\{\begin{array}{l} O_1(n+1) = (1-\mu)^{n+2}\cdot(X_1-X_2)+\left(\sum\limits_{k=0}^{n}[(1-\mu)^{k+1}\cdot(-1)^k]\right)\cdot X_{of2}+ \\ \left(-\sum\limits_{k=0}^{n}[(1-\mu)^{k+1}\cdot(-1)^k]+(1-\mu)\right)\cdot X_{of1}-(1-\mu)X_{of1}+X_{of1} \\ O_2(n+1) = X_{of2} \end{array}\right. \tag{B.10}$$

Making the change of variable $j = k+1$ and knowing that $(-1)^{j-1} = -(-1)^j$, Eq. (B.10) results in Eq. (B.11) after some simplifications, which is the same as the hypothesis Eq. (B.6).

$$\begin{array}{l} \left\{\begin{array}{l} O_1(n+1) = (1-\mu)^{n+2}\cdot(X_1-X_2)+\left(-\sum\limits_{j=1}^{n+1}[(1-\mu)^j\cdot(-1)^j]\right)\cdot X_{of2}+ \\ \left(\sum\limits_{j=1}^{n+1}[(1-\mu)^j\cdot(-1)^j]\right)\cdot X_{of1}+X_{of1} \\ O_2(n+1) = X_{of2} \end{array}\right. \Leftrightarrow \\ \Leftrightarrow\left\{\begin{array}{l} O_1(n+1) = (1-\mu)^{n+2}\cdot(X_1-X_2)+\left(-\sum\limits_{j=0}^{n+1}[(1-\mu)^j\cdot(-1)^j]+1\right)\cdot X_{of2}+ \\ \left(\sum\limits_{j=0}^{n+1}[(1-\mu)^j\cdot(-1)^j]\right)\cdot X_{of1} \\ O_2(n+1) = X_{of2} \end{array}\right. \end{array} \tag{B.11}$$

The same is true if the procedure is applied for n odd.

$\therefore$ The thesis is then proven to be true and equations (B.4) and (B.5) represent Eq. (B.1) and Eq. (B.2) for all n.

APPENDIX C
NONIDEAL SI LOCAL OFFSET CLOSE FORM OUTPUT EQUATION

This appendix presents the close form derivation for the SI gamma local offset canceling scheme of Chapter 3 with non-ideal gains. For convenience equations (3-17) and (3-18) are rewritten here in Eq. (C.1) and Eq. (C.2).

$$O_1(n) = \begin{cases} X_{of1} & ,n\ even \\ A_1 \cdot (1-\mu) \cdot [O_2(n-1) - \alpha O_1(n-1)] + X_{of1} & ,n\ odd \end{cases} \tag{C.1}$$

$$O_2(n) = \begin{cases} A_2(1-\mu) \cdot [O_1(n-1) - \alpha O_2(n-1)] + X_{of2} & ,n\ even \\ X_{of2} & ,n\ odd \end{cases} \tag{C.2}$$

To find a close form solution for these equation a few iterations have to be performed over Eq. (C.1) and Eq. (C.2). From these a close form can be intuited. With the result a proof by induction is sufficient to attest the validity of the close form.

Assume that the initial conditions are $O_1(-1) = X_1$, $O_2(-1) = X_2$ and that the time starts at $n = 0$. According to equations (C.1) and (C.2), the iterative Eq. (C.3) can be found.

It should be noted that it was considered in Eq. (C.1) and Eq. (C.2) that the final offsets, X_{ofT1} and X_{ofT2}, on the reference phase are approximately equal to X_{of1} and X_{of2}, as shown in Chapter 3.

The close form equations have to be considered for the sequences n odd and n even, consequently two hypothesis have to be tested, for n even and n odd. After some minor manipulations, the following thesis is proposed.

$$n = 0 = even \rightarrow \begin{cases} O_1(0) = X_{of1} \\ O_2(0) = A_2 \cdot (1-\mu) \cdot (X_1 - \alpha X_2) + X_{of2} \end{cases}$$

$$n = 1 = odd \rightarrow \begin{cases} O_1(1) = A_1 A_2 (1-\mu)^2 \cdot (X_1 - \alpha X_2) + A_1(1-\mu) \cdot X_{of2} - A_1(1-\mu)\alpha \cdot X_{of1} + X_{of1} \\ O_2(1) = X_{of2} \end{cases}$$

$$n = 2 = even \rightarrow \begin{cases} O_1(2) = X_{of1} \\ O_2(2) = A_1 A_2^2 (1-\mu)^3 \cdot (X_1 - \alpha X_2) + A_1 A_2 (1-\mu)^2 \cdot X_{of2} - A_1 A_2 (1-\mu)^2 \alpha \cdot X_{of1} + \\ A_2(1-\mu) \cdot X_{of1} - A_2(1-\mu)\alpha \cdot X_{of2} + X_{of2} \end{cases}$$

$$n = 3 = odd \rightarrow \begin{cases} O_1(3) = A_1^2 A_2^2 (1-\mu)^4 \cdot (X_1 - \alpha X_2) + A_1^2 A_2 (1-\mu)^3 \cdot X_{of2} - A_1^2 A_2 (1-\mu)^3 \alpha \cdot X_{of1} + \\ A_1 A_2 (1-\mu)^2 \cdot X_{of1} - A_1 A_2 (1-\mu)^2 \alpha \cdot X_{of2} + A_1(1-\mu) \cdot X_{of2} - A_1(1-\mu)\alpha \cdot X_{of1} + X_{of1} \\ O_2(3) = X_{of2} \end{cases} \tag{C.3}$$

$$n = 4 = even \rightarrow \begin{cases} O_1(4) = X_{of1} \\ O_2(4) = A_1^2 A_2^3 (1-\mu)^5 \cdot (X_1 - \alpha X_2) + A_1^2 A_2^2 (1-\mu)^4 \cdot X_{of2} - A_1^2 A_2^2 (1-\mu)^4 \alpha \cdot X_{of1} + \\ A_1 A_2^2 (1-\mu)^3 \cdot X_{of1} - A_1 A_2^2 (1-\mu)^3 \alpha \cdot X_{of2} + A_1 A_2 (1-\mu)^2 \cdot X_{of2} - A_1 A_2 (1-\mu)^2 \alpha \cdot X_{of1} + \\ A_2(1-\mu) \cdot X_{of1} - A_2(1-\mu)\alpha \cdot X_{of2} + X_{of2} \end{cases}$$

...

C.1 Close form proof by induction

C.1.1 Thesis

The close form of Eq. (C.1) and Eq. (C.2) is represented by Eq. (C.4) and Eq. (C.5).

$$n = even \rightarrow \begin{cases} O_1(n) = X_{of1} \\ O_2(n) = A_1^{(n/2)} A_2^{(n/2+1)} (1-\mu)^{n+1} \cdot (X_1 - \alpha X_2) + \\ \sum\limits_{k=0}^{n/2} [(1-\mu)^{2k} \cdot (A_1 A_2)^k] \cdot X_{of2} + \\ -\alpha \cdot \left(\sum\limits_{k=0}^{n/2} [(1-\mu)^{(2k+1)} \cdot A_1^k A_2^{(k+1)}] - (1-\mu)^{(n+1)} A_1^{(n/2)} A_2^{(n/2+1)} \right) \cdot X_{of2} + \\ \left(\sum\limits_{k=0}^{n/2} [(1-\mu)^{(2k+1)} \cdot A_1^k A_2^{(k+1)}] - (1-\mu)^{(n+1)} A_1^{(n/2)} A_2^{(n/2+1)} \right) \cdot X_{of1} + \\ -\alpha \cdot \left(\sum\limits_{k=0}^{n/2} [(1-\mu)^{2k} \cdot (A_1 A_2)^k] - 1 \right) \cdot X_{of1} \end{cases} \tag{C.4}$$

$$n = odd \rightarrow \begin{cases} O_1(n) = (A_1A_2)^{\left(\frac{n+1}{2}\right)} \cdot (1-\mu)^{(n+1)} \cdot (X_1 - \alpha X_2) + \\ \left(\sum_{k=1}^{\frac{n+1}{2}} [(1-\mu)^{(2k-1)} \cdot A_1^k A_2^{k-1}] - \alpha \cdot \left(\sum_{k=0}^{\frac{n-1}{2}} [(1-\mu)^{2k} \cdot (A_1A_2)^k] - 1\right)\right) \cdot X_{of2} + \\ \left(\sum_{k=0}^{\frac{n-1}{2}} [(1-\mu)^{2k} \cdot (A_1A_2)^k] - \alpha \cdot \left(\sum_{k=1}^{\frac{n+1}{2}} [(1-\mu)^{(2k-1)} \cdot A_1^k A_2^{(k-1)}]\right)\right) \cdot X_{of1} \\ O_2(n) = X_{of2} \end{cases} \tag{C.5}$$

C.1.2 Hypothesis

If n is even then Eq. (C.6) is observed:

$$n+1 = odd \rightarrow \begin{cases} O_1(n) = (A_1A_2)^{\left(\frac{n+2}{2}\right)} \cdot (1-\mu)^{(n+2)} \cdot (X_1 - \alpha X_2) + \\ \left(\sum_{k=1}^{\frac{n+2}{2}} [(1-\mu)^{(2k-1)} \cdot A_1^k A_2^{k-1}] - \alpha \cdot \left(\sum_{k=0}^{\frac{n}{2}} [(1-\mu)^{2k} \cdot (A_1A_2)^k] - 1\right)\right) \cdot X_{of2} + \\ \left(\sum_{k=0}^{\frac{n}{2}} [(1-\mu)^{2k} \cdot (A_1A_2)^k] - \alpha \cdot \left(\sum_{k=1}^{\frac{n+2}{2}} [(1-\mu)^{(2k-1)} \cdot A_1^k A_2^{(k-1)}]\right)\right) \cdot X_{of1} \\ O_2(n) = X_{of2} \end{cases} \tag{C.6}$$

Otherwise, if n is odd Eq. (C.7) is observed:

$$n+1 = even \rightarrow \begin{cases} O_1(0) = X_{of1} \\ O_2(n) = A_1^{\left(\frac{n+1}{2}\right)} A_2^{\left(\frac{n+1}{2}+1\right)} (1-\mu)^{(n+2)} \cdot (X_1 - \alpha X_2) + \\ \left(\sum_{k=0}^{\frac{n+1}{2}} [(1-\mu)^{2k} \cdot (A_1 A_2)^k] \right) \cdot X_{of2} + \\ -\alpha \cdot \left(\sum_{k=0}^{\frac{n+1}{2}} [(1-\mu)^{(2k+1)} \cdot A_1^k A_2^{(k+1)}] - (1-\mu)^{(n+2)} A_1^{((n+1)/2)} A_2^{((n+1)/2+1)} \right) \cdot X_{of2} \\ \left(\sum_{k=0}^{\frac{n+1}{2}} [(1-\mu)^{(2k+1)} \cdot A_1^k A_2^{(k+1)}] - A_1^{((n+1)/2)} A_2^{((n+1)/2+1)} \right) \cdot X_{of1} \\ -\alpha \cdot \left(\sum_{k=0}^{\frac{n+1}{2}} [(1-\mu)^{2k} \cdot (A_1 A_2)^k] - 1 \right) \cdot X_{of1} \end{cases} \quad \text{(C.7)}$$

C.1.3 Proof of the hypothesis veracity

- The first elements of the series are $n = 0$ and $n = 1$. Taking these values into equations (C.4) and (C.5), respectively, results in:

$$\begin{cases} O_1(0) = X_{of1} \\ O_2(0) = A_2(1-\mu) \cdot (X_1 - \alpha X_2) + X_{of2} \end{cases} \quad \text{(C.8)}$$

$$\begin{cases} O_1(1) = A_1 A_2 (1-\mu)^2 \cdot (X_1 - \alpha X_2) + A_1(1-\mu) \cdot X_{of2} + X_{of1} - \alpha A_1 (1-\mu) \cdot X_{of1} \\ O_2(1) = X_{of2} \end{cases} \quad \text{(C.9)}$$

This result matches the one for $n = 0$ and $n = 1$ in (C.4).

∴ Equations (C.4) and (C.5) are representations for the first element of (C.1) and (C.2)

- If n is even then $n + 1$ is odd. Taking Eq. (C.4) and substituting in Eq. (C.1) and Eq. (C.2) for n odd results in:

$$n = even \rightarrow \begin{cases} O_1(n) = X_{of1} \\ O_2(n) = (A_1A_2)^{(n/2+1)}(1-\mu)^{n+2} \cdot (X_1 - \alpha X_2) + \\ \left(\sum_{k=0}^{n/2} [(1-\mu)^{(2k+1)} \cdot A_1^{(k+1)}A_2^{k}]\right) \cdot X_{of2} + \\ -\alpha \cdot \left(\sum_{k=0}^{n/2} [(1-\mu)^{(2k+2)} \cdot (A_1A_2)^{(k+1)}] - (1-\mu)^{(n+2)}(A_1A_2)^{(n/2+1)}\right) \cdot X_{of2} \\ \left(\sum_{k=0}^{n/2} [(1-\mu)^{(2k+2)} \cdot (A_1A_2)^{(k+1)}] - (1-\mu)^{(n+2)}(A_1A_2)^{(n/2+1)}\right) \cdot X_{of1} \\ -\alpha \cdot \left(\sum_{k=0}^{n/2} [(1-\mu)^{(2k+1)} \cdot A_1^{(k+1)}A_2^{k}] - 1\right) \cdot X_{of1} \end{cases} \qquad (C.10)$$

making the change of variable $j = k + 1$, equation (C.10) and after some simplifications results in (C.11), which is the same as the hypothesis equation (C.6).

$$n+1=odd\rightarrow\begin{cases}O_1(n)=X_{of1}\\ O_2(n)=(A_1A_2)^{\left(\frac{n+2}{2}\right)}(1-\mu)^{(n+2)}\cdot(X_1-\alpha X_2)+\\ \left(\sum\limits_{j=1}^{n/2+1}[(1-\mu)^{(2j-1)}\cdot A_1^jA_2^{j-1}]\right)\cdot X_{of2}-\\ \alpha\cdot\left(\sum\limits_{j=1}^{n/2+1}[(1-\mu)^{2j}\cdot(A_1A_2)^j]-(1-\mu)^{(n+2)}(A_1A_2)^{(n/2+1)}\right)\cdot X_{of2}+\\ \left(\sum\limits_{j=1}^{n/2+1}[(1-\mu)^{2j}\cdot(A_1A_2)^j]-(1-\mu)^{(n+2)}(A_1A_2)^{(n/2+1)}\right)\cdot X_{of1}-\\ \alpha\cdot\left(\sum\limits_{j=1}^{n/2+1}[(1-\mu)^{(2j-1)}\cdot A_1^jA_2^{j-1}]-A_1(1-\mu)\right)\cdot X_{of1}-\alpha A_1(1-\mu)X_{of1}+X_{of1}\end{cases}\tag{C.11}$$

$$\Leftrightarrow$$

$$n+1=odd\rightarrow\begin{cases}O_1(n)=X_{of1}\\ O_2(n)=(A_1A_2)^{\left(\frac{n+2}{2}\right)}(1-\mu)^{(n+2)}\cdot(X_1-\alpha X_2)+\\ \left(\sum\limits_{j=1}^{\frac{n+2}{2}}[(1-\mu)^{(2j-1)}\cdot A_1^jA_2^{j-1}]-\alpha\cdot\left(\sum\limits_{j=0}^{n/2}[(1-\mu)^{2j}\cdot(A_1A_2)^j]-1\right)\right)\cdot X_{of2}+\\ \left(\sum\limits_{k=0}^{n/2}[(1-\mu)^{2j}\cdot(A_1A_2)^j]-\alpha\left(\sum\limits_{k=1}^{\frac{n+2}{2}}[(1-\mu)^{(2j-1)}\cdot A_1^jA_2^{j-1}]\right)\right)\cdot X_{of1}+\end{cases}$$

The very same procedure can be applied to n even and the same conclusion is reached

∴ The thesis is then proven to be true and equations (C.4) and (C.5) represent Eq. (C.1) and Eq. (C.2) for all n.

C.2 Limit results.

The summing parcels of Eq. (C.4) and Eq. (C.5) can be recognized as the sum of a geometric progression terms. Consequently when $n\rightarrow\infty$ these equations will converge if and only if the following inequality holds:

$$A_1A_2(1-\mu)^2<1 \tag{C.12}$$

Under this condition it is not difficult to conclude that only the terms in the summing parcels will tend to a value different from zero. All other isolated terms will go to zero. Then the limit is found to be:

$$\begin{gathered}
(n=odd)\rightarrow\begin{cases} O_1(n) = \left((1-\mu)A_1\dfrac{1-\alpha(1-\mu)A_2}{1-(1-\mu)^2A_1A_2}\right)\cdot X_{of2}+ \\ \left(\dfrac{1-\alpha(1-\mu)A_1}{1-(1-\mu)^2A_1A_2}\right)\cdot X_{of1} \\ O_2(n)\equiv X_{of2} \end{cases} \\
(n=even)\rightarrow\begin{cases} O_1(n)\equiv X_{of1} \\ O_2(n) = \left(\dfrac{1-\alpha(1-\mu)A_2}{1-(1-\mu)^2A_1A_2}\right)\cdot X_{of2}+ \\ \left((1-\mu)A_2\dfrac{1-\alpha(1-\mu)A_1}{1-(1-\mu)^2A_1A_2}\right)\cdot X_{of1} \end{cases}
\end{gathered} \tag{C.13}$$

APPENDIX D
PROOF BY INDUCTION OF F&H x(t) EQUATION

Equation (D.1) is the general form representation of a system with a given sate-space representation. Equation (D.2) is to be proven true for $\forall n \in \aleph \cup \{0\}$.

$$\begin{cases} x(t) = e^{A \cdot (t - t_0)} \cdot x(t_0) + \int_{t_0}^{t} (e^{A \cdot (t - \alpha)} \cdot B \cdot u_s(\alpha)) d\alpha \\ \\ y(t) = C \cdot x(t) + D \cdot u_s(t) \end{cases} \tag{D.1}$$

$$\begin{cases} x((n-1)T + \tau) = \sum_{i=0}^{n-1} u_i \cdot (e^{A \cdot \tau})^{n-1-i} \cdot \left(\int_0^{\tau} (e^{A \cdot (\tau - \beta)}) d\beta \right) \cdot B & \forall n \in \aleph \\ x(0^+) = 0 & n = 0 \\ y((n-1)T + \tau) = C \cdot x((n-1)T + \tau) + D \cdot u_n \end{cases} \tag{D.2}$$

$$Dn = \{ t \in \Re : (n-1)T + \tau \le t \le nT, \ \forall n \in \aleph \}$$

$$(x(t) = x((n-1)T + \tau)), (y(t) = y((n-1)T + \tau)) \ \ \forall t \in Dn, \forall n \in \aleph$$

D.1 Hypothesis

If Eq. (D.2) is valid for all n, (in particular for $x(t)|_{t-nT}$), then for $n + 1$ Eq. (D.3) is true.

$$\begin{cases} x(nT+\tau) = \sum_{i=0}^{n} u_i \cdot (e^{A\cdot\tau})^{n-i} \cdot \left(\int_0^{\tau} (e^{A\cdot(\tau-\beta)}) d\beta \right) \cdot B & \forall n \in \aleph \\ x(0^+) = 0 & n = 0 \\ y(nT+\tau) = C \cdot x(nT+\tau) + D \cdot u_n \end{cases} \tag{D.3}$$

$$Dn = \{ t \in \Re : nT + \tau \le t \le (n+1)T,\ \forall n \in \aleph \}$$
$$(x(t) = x(nT+\tau)), (y(t) = y(nT+\tau))\ \ \forall t \in Dn, \forall n \in \aleph$$

D.2 Proof of the hypothesis veracity

- The first element of the series is when $n = 1$. Then directly from Eq. (D.2):

$$\begin{cases} x(\tau) = u_0 \cdot \left(\int_0^{\tau} (e^{A\cdot(\tau-\beta)}) d\beta \right) \cdot B \\ x(0^+) = 0 & n = 0 \\ y(\tau) = C \cdot x(\tau) + D \cdot u_n \end{cases} \tag{D.4}$$

$$D1 = \{ t \in \Re : \tau \le t \le T \}$$
$$(x(t) = x(\tau)), (y(t) = y(\tau))\ \ \forall t \in D1$$

Using Eq. (D.1), and assuming initial state $x(t)|_{t_0 = 0^-} = 0$, results in:

$$x(\tau) = u_0 \cdot \left(\int_0^{\tau} (e^{A\cdot(\tau-\beta)}) d\beta \right) \cdot B \tag{D.5}$$

Since the integration is halted for the rest of the period $T - \tau$ (s), then $x(t)$ is constant during the interval of time $t \in [\tau, T]$ (s). This result can be represented as Eq. (D.4).

$\therefore$ Equation (D.2) gives a correct result for the first element of the series.

- To prove for $n+1$ equation (D.1) needs to be calculated for $t = nT+\tau$. Using the initial condition $x(t)\big|_{t_0 = nT^-} = x(nT)$ given by Eq. (D.2). results in the following equation:

$$x(nT+\tau) = e^{A\cdot(nT+\tau-nT)}\cdot\sum_{i=0}^{n-1}u_i\cdot(e^{A\cdot\tau})^{n-1-i}\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\beta)})d\beta\right)\cdot B+$$
$$+\int_{nT}^{nT+\tau}(e^{A\cdot(nT+\tau-\alpha)}\cdot B\cdot u_s(\alpha))d\alpha \tag{D.6}$$

Making the change of variable $\beta = \alpha - T$ on the second integral and knowing that $u_s(\alpha) = u_n, \alpha\in[nT,nT+\tau]$ then results into Eq. (D.7)

$$x(nT+\tau) = \left(\sum_{i=0}^{n-1}u_i\cdot(e^{A\cdot\tau})^{n-i}+u_n\right)\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\beta)})d\beta\right)\cdot B$$
$$\Leftrightarrow x(nT+\tau) = \sum_{i=0}^{n}u_i\cdot(e^{A\cdot\tau})^{n-i}\cdot\left(\int_0^{\tau}(e^{A\cdot(\tau-\beta)})d\beta\right)\cdot B \tag{D.7}$$

Having in mind that the integration is halted for $t\in[nT+\tau, nT]$, $\forall n\in\aleph$ then Eq. (D.7) can be represented by Eq. (D.3).

$\therefore$ The thesis is then proven to be true and Eq. (D.2) represent Eq. (D.1) for all n.

APPENDIX E
DISCRETE-TIME F&H GENERIC DIFFERENCE EQUATION AND Z TRANSFORM

$$x(nT) = e^{A\tau} \cdot x[(n-1)T] + u_{n-1} \cdot \left(\int_0^{\tau} e^{A(\tau - \alpha)} d\alpha \right) \cdot B \qquad (E.1)$$

$$\Leftrightarrow x(nT) = e^{A\tau} \cdot x[(n-1)T] + u_{n-1} \cdot \left(-A^{-1} e^{A(\tau-\alpha)} \Big|_0^{\tau} \right) \cdot B$$

$$\Leftrightarrow x(nT) = e^{A\tau} \cdot x[(n-1)T] + u_{n-1} \cdot [-A^{-1} \cdot I + A^{-1} \cdot e^{A\tau}] \cdot B$$

$$\Leftrightarrow x(nT) = e^{A\tau} \cdot x[(n-1)T] + u_{n-1} \cdot [e^{A\tau} - I] \cdot A^{-1} \cdot B \qquad (E.2)$$

$$Z\{\ x(nT) = e^{A\tau} \cdot x[(n-1)T] + u_{n-1} \cdot [e^{A\tau} - I] \cdot A^{-1} \cdot B\} \rightarrow$$

$$X(z) - z^{-1} \cdot e^{A\tau} \cdot X(z) = z^{-1} U(z) \cdot [e^{A\tau} - I] \cdot A^{-1} \cdot B$$

$$\Leftrightarrow X(z) = [I - z^{-1} \cdot e^{A\tau}]^{-1} \cdot z^{-1} U(z) \cdot [e^{A\tau} - I] \cdot A^{-1} \cdot B$$

$$Y(z) = C \cdot X(z) + D \cdot U(z)$$

$$\Leftrightarrow Y(z)/U(z) = C \cdot [I - z^{-1} \cdot e^{A\tau}]^{-1} \cdot z^{-1} \cdot [e^{A\tau} - I] \cdot A^{-1} \cdot B + D$$

$$\Leftrightarrow Y(z)/U(z) = (C \cdot [I - z^{-1} \cdot e^{AkT}]^{-1} \cdot z^{-1} \cdot [e^{AkT} - I] \cdot A^{-1} \cdot B + D); k = \tau/T \qquad (E.3)$$

REFERENCES

[1] W. Freeman, *Mass action in the nervous system.* New York, NY: Academic Press, 1975.

[2] W. Freeman, "Tutorial on neurobiology from single neurons to brain chaos," *International Journal of Biforcation and Chaos*, 2(3), pp. 451–482, 1992.

[3] Y. Yao, and W. Freeman, "Model of bological pattern recognition with spatially chaotic dynamics," *Neural Networks*, 3, pp. 153–170, 1990.

[4] W. McCulloch, and W. Pitts, "A Logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, 5, pp. 115–133, 1943.

[5] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, 65, pp. 386–408,1958.

[6] R. Hecht-Nielsen, *Neurocomputing.* Addison-Wesley, 1990.

[7] S. Haykin, *Neural Networks: A comprehensive Foundation.* Macmillan College Publishing, 1994.

[8] E. Lorenz "Deterministic nonperiodic flow," *Journal of Atmospheric Science*, 20, pp. 130, 1963.

[9] T. Parker, and L. Chua, "Chaos: a tutorial for engineers," *Proc. IEEE*, 75(8), pp. 982–1008, Aug. 1987.

[10] L. Shilnikov, "Strange attractors and dynamical models," *IEEE Journal of Circuits, systems and Computers*, 3, pp. 1–10, 1993.

[11] S. Strogatz, *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering.* Addison-Wesley, 1994.

[12] J. Hopfield, "Neural Networks and physical systems with emergent collective computational abilities," *Proc. of the National Academy od Sciences*, 79, pp. 2554–2558, 1982.

[13] J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. of the National Academy od Sciences*, 81, pp. 3088–3092, 1984.

[14] B. Katz, *Nerve, muscle and synapse.* McGraw-Hill, 1966.

[15] W. Freeman, and K. Grajski, "Relation of olfactory EEG to behavior: Factor analysis," *Behavioral Neuroscience*, 101(6), pp. 766–777, 1987.

[16] W. Freeman, Y. Yao, and B. Burke, "Central pattern generating and recognizing on olfactory bulb: A correlation learning rule," *Neural Networks*, 1, pp. 277-288, 1988.

[17] Y. Yao, W. Freeman, B. Burke, and Q. Yang, "Pattern recognition by a distributed neural network: an industrial application," *Neural Networks*, 4, pp. 103-121, 1991.

[18] K. Shimoid and W. Freeman, "Dynamic Neural Network Derived from the Olfactory System with Examples of Applications", *IEICE Trans. Fundamentals*, E78(7), pp. 869–884, Jul. 1995.

[19] W. Freeman, H. Chang, B. Burke, P. Rose, and J. Badler "Taming chaos: Stabilization of aperiodic attractors by noise," *IEEE Trans. Circuits and Systems*, Oct. 1997.

[20] H. Chang, W. Freeman, and B. Burke "Biologically modeled noise stabilizing neurodynamics for pattern recognition," To appear in the International Journal of Bifurcation and Chaos.

[21] K. Wiesenfeld, and P. Hadley, "Attractor crowding in oscillator arrays," *Physics Review Letters*, 62, pp. 1335–1338, 1989.

[22] K. Tsang, and K. Wiesenfeld, "Attractor crowding in Josephson junction arrays," *Applied Physics Letters*, 56, pp. 4745–4749, 1990.

[23] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical recipes: The art of scientific computing.* New York, NY: Cambridge University press, 1986.

[24] J. Principe, V. Tavares, J. Harris, and W. Freeman, "Design and implementation of a biologically realistic olfactory cortex in analog VLSI," *Proc. IEEE*, 89(7),pp. 1030–1051, Jul. 2001.

[25] V. Tavares, J. Principe, and G. Lynn, "Discrete network implementation of Freemanís model of the olfactory system," *Int. Join Conference Neural Networks*, 2, pp. 1488–1493, 1998.

[26] L. Rabiner, and R. Schafer, *Digital processing of speech signals.* Englewood Cliffs, NJ: Prentice-Hall, 1978.

[27] K. Steiglitz, "The equivalence of digital and analog signal processing," *Information and Control*, 8, pp. 455–467, 1965.

[28] D. Amit, *Modeling brain function.* New York, NY: Cambridge University Press, 1989.

[29] P. Peretto, *An introduction to the modeling of neural networks.* New York, NY: Cambridge University Press, 1994.

[30] A. Oppenheim, and R. Schafer *Discrete-time signal processing.* Englewood Cliffs, N J: Prentice-Hall, 1989.

[31] J. Principe, B. de Vries, and P. Oliveira, "The gamma filter a new class of adaptive IIR filters with restricted feedback", *IEEE Trans. Signal Processing*,41(2), pp. 649–656, Feb. 1986.

[32] B. de Vries, *Temporal processing with neural networks: The development of the gamma model.* University of Florida FL: PhD. Dissertation, 1991.

[33] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proc. IEEE*, 78, pp. 1550–1560, 1990.

[34] M. Abeles, *Corticonics.* New York, NY: Cambridge University Press, 1991.

[35] G. Goodwin, and K. Sin, *Adaptive filtering prediction and control.* Englewood Cliffs, NJ: Prentice Hall, 1984.

[36] Neurosolutions, *Neurosolutions manual.* Gainesville FL: NeuroDimentions Incorporated, 1995.

[37] Matlab *Users manual.* Cambridge, MA: Mathworks, 1998.

[38] W. Freeman, "Mesoscopic neurodynamics: from neuron to brain," *Journal of Physiology*, 2000 (in Press).

[39] J. L. McCreaty, and P. R. Gray, "All-MOS charge redistribution analog-to-digital conversion techniques - part I," *IEEE Journal of Solid-State Circuits*, SC-10(6), pp. 371–379, Dec. 1975.

[40] R. Suarez, P. Gray, and, D. Hodges, "All-MOS charge redistribution analog-to-digital conversion techniques - part II," *IEEE Journal of Solid-State Circuits*, SC-10(6), pp. 379–385, Dec. 1975.

[41] R. W. Brodersen, P. R. Gray, and D. A. Hodges "MOS switched-capacitor filters," *Proc. IEEE*, 67, pp. 61–75, Jan. 1979.

[42] J. Hughes, N. Bird, and I. Macbeth, "Switched currents—A new tecnique for analog sampled-data signal processing", *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 1584–1587, 1989.

[43] C. Toumazou, F. J. Lidgey, and D. G. Haigh, *Analogue IC design: The current-mode approach.* London, UK: Peter Peregrinus,1990.

[44] M. Ismail, and T. Fiez, *Analog VLSI: Signal and information processing*. McGraw-Hill International editions—Electronic Engineering Series, 1994.

[45] E. Seevink, and R. J. Wiegrink, "Generalized translinear circuit principle," *IEEE Journal of Solid-State Circuits*, 26(8), pp. 1098–1102, Aug. 1991.

[46] K. Bult, and H. Wallinga, "A class of analog CMOS circuits based on the square-law characteristic of an MOS transistor in saturation," *IEEE Journal of Solid-State Circuits*, SC-22(3), pp. 357–365, Jun.1987.

[47] B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response," *IEEE Journal of Solid-State Circuits*, SC-3(4), pp. 365–373, Dec.1968.

[48] R. J. Wiegerink, "A CMOS four-quadrant analog current multiplier," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 2244–2247, 1991.

[49] M. H. Cohen, and A. G. Andreou, "Current-mode subthreshold MOS implementation of the Herault-jutten autoadaptive network," *IEEE Journal of Solid-State Circuits*, 27(5), pp. 714–727, May.1992.

[50] V. Tavares, and J. Principe, "Towards a silicon smplementation of the slfactory system," *IEEE Int. Conf. Circuits and Systems*, 2, pp. 131–134, 1998.

[51] J. Nedved, J. Vanneuville, D. Gevaert, and J. Sevenhans, "A transistor-only switched current sigma-delta A/D converter for a CMOS speech CODEC," *IEEE Journal of Solid-State Circuits*, 30(7), pp. 819–822, Jul. 1995.

[52] K. Koli, and K. Halonen, "A fully differential class-AB switched-current integrator for signal processing," *IEEE Journal of Solid-State Circuits*, 32(2), pp. 238–244, Feb. 1997.

[53] N. C. Battersby, and C. Toumazou, "Class AB switched-current memory for analog sampled-data systems," *Ellectronics Letters*, 27(10), pp. 873–874, May. 1991.

[54] HSPICE, *Users manual.* Campbell, CA: Meta-Software, 1992.

[55] W. B. Wilson, H. Z. Massoud, E. J. Swanson, R. George Jr, and R. B. Fair, "Measurement and modeling of charge feedthrough in n-channel MOS analog switches," *IEEE Journal of Solid-State Circuits*, SC-20(6), pp. 1206–1213, Dec. 1985.

[56] B. J. Sheu, and C. Hu, "Switch-induced error voltage on a switched capacitor," *IEEE Journal of Solid-State Circuits*, SC-19(4), pp. 519–525, Aug. 1984.

[57] G. Wegmann, E. A. Vittoz, and F. Rahali, “Charge injection in analog switches,” *IEEE Journal of Solid-State Circuits*, SC-22(6), pp. 1091–1097, Dec. 1987.

[58] V. Tavares, J. Principe, and J. Harris, “The F&H filter: a novel ultra low power discrete time filtering technique,” *IEE Electronics Letters*, 35(15), pp. 1226–1227, 1999.

[59] D. Lencaster, *Active-filter cookbook*, Indianapolis:IN, Howard W. Sams, 1975.

[60] R. C. Dorf, and R. H. Bishop, *Modern Control Systems,* Menlo Park, Ca: Addison Wesley Longman INC., 1998.

[61] T. Kailath, *Linear Systems.* Englewood Cliffs, N J: Prentice Hall, 1980.

[62] N. K. Sinha, and G. P. Rao, *Identification of continous-time systems—Methodology and computer implementation.* Kluwer Academic Publishers, 1992.

[63] S. Franco, *Design with Operational Amplifiers and analog integrated circuits.* McGraw-Hill, Inc., 1988.

[64] J. Franca, and Y. Tsividis, *Design of analog-digital VLSI circuits for telecomunications and signal processing.* Englewood Cliffs, NJ: Prentice Hall, 1994.

[65] C. Mead, *Analog VLSI neural systems.* Addison-Wesley, 1989.

[66] V. Tavares, J. Principe, and J. Harris, “A silicon olfactory bulb oscillator,” *International Symposium on Circuits and Systems*, pp. 397–400, 2000.

[67] J. Eisenberg, W. Freeman, and B Burke, “Hardware architecture of a neural network model simulating pattern recognition by the olfactory bulb,” *Neural Networks*, 2, pp. 315–325, 1989.

[68] D. Rumelhart, G. Hinton, and R. Williams, “Learning representations by backpropagating errors,” *Nature*, 323, pp. 533–536,1986.

[69] Y. Tsividis, M. Banu, and J. Khoury “Continous-time MOSFET-C filters in VLSI,” *IEEE Journal of Solid-State Circuits*, SC-21(1), pp. 15–30, Feb. 1986.

[70] C. Maxwell, *A treatise on electricity and magnetism.* London, UK: Low and Brydone, 1946.

[71] J. D. L. Fried, “Analog sampled-data filters,” *IEEE Journal of Solid-State Circuits*, SC-7(4), pp. 302–304, Aug. 1972.

[72] D. A. Johns, and Ken Martin, *Analog integrated circuit design.* John Wiley & Sons, INC., 1997.

[73] D. R. Frey, "Log domain filtering for RF applications," *IEEE Journal of Solid-State Circuits*, SC-21(1), pp. 15–30, Feb. 1986.

[74] D. R. Frey, "Log domain filtering: An approach to current mode filtering," *IEE proc., Pt. G*, 140, pp. 406–416, Dec. 1993.

[75] D. R. Frey, "Exponential state space filters: A generic current mode design strat- *Trans. Circuits and Systems I*, 43(1), pp. 34–42, Jan. 1996.

[76] R. H. Zele, and D. J. Allstot, "Low power CMOS continuous-time filters," *IEEE Journal of Solid-State Circuits*, 31(2), pp. 157–168, Feb. 1996.

[77] M. Banu, and Y. Tsividis, "An elliptic continous-time CMOS filter with on-chip automatic tuning," *IEEE Journal of Solid-State Circuits*, SC-20(6), pp. 1114–1121, Dec. 1985.

[78] R. Gregorian, and G. C. Temes, *Analog MOS integrated circuits for signal processing*. John Wiley & Sons, INC., 1986.

[79] R. L. Geiger, P. E. Allen, and N. R. Strader, *VLSI design techniques for analog and digital circuits.* McGraw-Hill International editions—Electronic Engineering Series, 1990.

[80] S. Hayken, and B. Veen, *Signals and Systems.* John Wiley & Sons, 1999.

[81] R. J. Baker, H. W. Li, and D. E. Boyce, *CMOS circuit design, layout and simulation.* IEEE press, 1998.

[82] G. M. Jacobs, D. J. Allstot, R. W. Brodersen, and P. R. Gray, "MOS switched-capacitor ladder filters," *IEEE Trans. circuits and systems*, CAS-25, pp. 1014–1021, Dec. 1978.

[83] K. Martin, and A. S. Sedra, "Stray-insensitive switched-capacitor filters based on the bilinear z transform," *Electronic letters*, 19, pp. 365–366, 1979.

[84] K. Martin, "Improved circuits for the realization of switched-capacitor filters," *IEEE trans. circuits and systems*, CAS-27, pp. 237–244, Apr. 1980.

[85] Y. P. Tsividis, *Operation and modeling of the MOS transistor.* McGraw-Hill International editions—Electronic Engineering Series, 1988.

[86] C. Eichenberger, and W. Guggenbuhl, "Dummy transistor compensation of analog MOS switches," *IEEE Journal of Solid-State Circuits*, 24(4), pp. 1143–1145, Aug. 1989.

[87] C. Eichenberger, and W. Guggenbuhl, "On charge injection in analog MOS switches and dummy compensation techniques," *IEEE trans. Circuits and Systems*, CAS-37(2), pp. 256–264, Feb. 1990.

[88] K. C. Hsieh, P. R. Gray, D. Senderowicz, and D. G Messerschmitt, "A low-noise chopper-stabilized differential switched-capacitor filtering technique," *IEEE Journal of Solid-State Circuits*, SC-16, pp. 708–715, Dec. 1981.

[89] M. Nagata, "Limitations, innovations, and challenges of circuits and devices into a half micrometer and beyond," *IEEE Journal of Solid-State Circuits*, 27(4), pp. 465–472, Apr. 1992.

[90] R. R. Harrison, J. A. Bragg, P. Hasler, B. A. Minch, and S. Deweerth, "A CMOS programmable analog memory-cell array using floating-gate circuits," *IEEE trans. circuits and systems II*, 48(1), pp. 4–11, Jan. 2001.

[91] Y. Berg, and T. S. Lande, "Area effecient circuit tunning with floating-gate techniques," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 396–399, 1989

[92] K. C. Smith, and A. S. Sedra, "The current conveyor - A new circuit building block," *Proc. IEEE*, 56, pp. 1368–1369, Aug. 1968.

[93] A. S. Sedra, and K. C. Smith "A second generation current conveyor and its applications," *IEEE Trans. Circuits Theory*, CT-17, pp. 132–134, Feb. 1970.

[94] A. S. Sedra, G. W. Roberts and F. Gohh, "The current conveyor: history progress and new results," *IEE Proc. pt G*, 137, pp. 78–87, Apr. 1990.

[95] M. Bracey, W. Redman-White, J. Richardson, and J. B. Hughes, "A full Nyquist 15 MS/s 8-b differential switched-current A/D converter," *IEEE Journal of Solid-State Circuits*, 31(7), pp. 945–951, Jul. 1996.

[96] D. M. W. Leenaerts, G H. M. Joordeens, and J. A. Hegt, "A 3.3V 625kHz switched-current multiplier," *IEEE Journal of Solid-State Circuits*, 31(9), pp. 1340–1343, Sep. 1996.

[97] T. Reimann, F. Krummenacher, and M. J. Declercq, "An 8-b, 40 Msamples/s Switched-current-mode track-and-hold circuit on a biCMOS sea-of-gates array," *IEEE Journal of Solid-State Circuits*, 31(9), pp. 1340–1343, Mar. 1996.

[98] S. J. Daubert, D. Vallancourt and Y. P. Tsividis, "Current copier cells," *Ellectronic Letters*, 24, pp. 1560–1562, Dec. 1988.

[99] D. Vallancourt, and Y. P. Tsividis, "Sampled-current circuits," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 1592–1595, 1989.

[100] T. Fiez, and D. J. Allstot, "CMOS switched-current ladder filters," *IEEE Journal of Solid-State Circuits*, 25(6), pp. 1360–1367, Dec.1990.

[101] M. J. M. Pelgrom, A.C. J. Duinmaijer, and A. P. G. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, 24(5), pp. 1433–1439, Oct.1989.

[102] H. C. Yang, T. S. Fiez, and D. J. Allstot, "Current feedthrough effects and cancellation techniques in switched current circuits," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 3186–3188, 1990.

[103] J. B. Shyu, G. C. Temes, and F. Krummenacher, "Random error effects in matched MOS capacitors and current sources," *IEEE Journal of Solid-State Circuits*, SC-19(6), pp. 948–955, Dec.1984.

[104] U. Gatti, F. Maloberti, and V. Liberali, "Full stacked layout of analogue cells," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 1123–1126, 1989.

[105] T. S. Fiez, G. Liang, and D. J. Allstot, "Switched-current circuit design issues," *IEEE Journal of Solid-State Circuits*, 26(3), pp. 192–302, Mar.1991.

[106] J. B. Hughs, I. C. Macbeth, and D. M. Pattullo, "Second generation switched-current processing," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 2805–2808, 1990.

[107] G. Wegmann, and E. A. Vittoz, "Analysis and improvement of accurate dynamic current mirrors," *IEEE Journal of Solid-State Circuits*, 25(3), pp. 699–706, Jun.1990.

[108] M. Song, and Y. Lee, W. Kim, "A clock feedthrough reduction circuit for switched-current systems," *IEEE Journal of Solid-State Circuits*, 28(2), pp. 133–137, Feb. 1993.

[109] J. B. Hughs, and K. W. Moulding, "S^2I: A switched-current technique for high performance," *Ellectronic Letters*, 29, pp. 1400–1401, Aug. 1993.

[110] C. Toumazou, and S. Xiao, "n-step charge injection cancellation sheme for very accurate switched current circuits," *Ellectronics Letters*, 30, pp. 680–681, Apr. 1994.

[111] J. B. Hughes, and K. W. Moulding, "Switched-current signal processing for video frequencies and beyond," *IEEE Journal of Solid-State Circuits*, 28(3), pp. 314–322, Mar. 1993.

[112] J. B. Hughes, K. W. Moulding, J. Richardson, J. Bennett, W. Redman-white, M. Bracey, and R. S. Soin, "Automated design of switched-current filters," *IEEE Journal of Solid-State Circuits*, 31(7), pp. 898–907, Mar. 1993.

[113] X. Hu, and K. W. Martin, "A switched-current sample-and-hold circuit," *IEEE Journal of Solid-State Circuits*, 32(6), pp. 898–904, Jun. 1997.

[114] A.Worapishet, J. Hughes, and C. Toumazou, "An improved CMOS offset-compensated current comparator for high speed applications," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 535–538, 1998.

[115] D. G. Nairn, "Amplifiers for high-speed current-mode sample-and-hold circuits," *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 2045–2048, 1992.

[116] J. Fischer, "High-performance CMOS power amplifier," *IEEE Journal of Solid-State Circuits*, SC-20(6), pp. 1200–1205, Dec. 1985.

BIOGRAPHICAL SKETCH

Vitor Manuel Grade Tavares was born in June 5, 1968 in Vila Nova de Poiares, Portugal. He earned his *Licenciatura* and MS degrees in Electronics and Telecommunications Engineering from University of Aveiro, Portugal in 1991 and 1994 respectively. He has participated in different projects, such as the development of infrared data and speech diffusion, measurement and characterization of artificial light in infrared communication systems, real time ECG acquisition and visualization, microprocessor, SC, SI designs and other analog VLSI circuits for signal processing.

Since 1995 he has been a Ph.D. student at the Department of Electrical and Computer Engineering, in the CNEL Computational NeuroEngineering Laboratory at the University of Florida. His present interests are in the area of neuromorphic chip design and biomimetic computing.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Jose C. Principe, Chairman
Professor of Electrical and Computer
Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

John G. Harris

John G. Harris, Co-Chairman
Associate Professor of Electrical and
Computer Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Haniph Latchman
Associate Professor of Electrical and
Computer Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Purvis Bedenbaugh
Assistant Professor of Neuroscience

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Robert M. Fox
Associate Professor of Electrical and
Computer Engineering

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December, 2001

Pramod P. Khargonekar
Dean, College of Engineering

Winfred M. Phillips
Dean, Graduate School